TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Jenni Hukkanen

Contributions to Medical Image Segmentation and Signal Analysis Utilizing Model Selection Methods

Julkaisu 1548 • Publication 1548

Tampere 2018

Tampereen teknillinen yliopisto. Julkaisu 1548
Tampere University of Technology. Publication 1548

Jenni Hukkanen

Contributions to Medical Image Segmentation and Signal Analysis Utilizing Model Selection Methods

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB219, at Tampere University of Technology, on the 25th of May 2018, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2018

Doctoral candidate: Jenni Hukkanen
Laboratory of Signal Processing
Faculty of Computing and Electrical Engineering
Tampere University of Technology
Finland

Supervisor: Ioan Tabus, Prof.
Laboratory of Signal Processing
Faculty of Computing and Electrical Engineering
Tampere University of Technology
Finland

Pre-examiner: Radu Ciprian Bilcu, Dr.
Huawei Technologies Finland Oy
Finland

Pre-examiner and Opponent: Daniel Nicorici, Dr.
Orion Oyj
Finland

Opponent: Cristian Perra, Prof.
Department of Electrical and Electronic Engineering
University of Cagliari
Italy

ISBN 978-952-15-4142-1 (printed)
ISBN 978-952-15-4161-2 (PDF)
ISSN 1459-2045

Abstract

This thesis presents contributions to model selection techniques, especially based on information theoretic criteria, with the goal of solving problems appearing in signal analysis and in medical image representation, segmentation, and compression.

The field of medical image segmentation is wide and is quickly developing to make use of higher available computational power. This thesis concentrates on several applications that allow the utilization of parametric models for image and signal representation. One important application is cell nuclei segmentation from histological images. We model nuclei contours by ellipses and thus the complicated problem of separating overlapping nuclei can be rephrased as a model selection problem, where the number of nuclei, their shapes, and their locations define one segmentation. In this thesis, we present methods for model selection in this parametric setting, where the intuitive algorithms are combined with more principled ones, namely those based on the minimum description length (MDL) principle. The results of the introduced unsupervised segmentation algorithm are compared with human subject segmentations, and are also evaluated with the help of a pathology expert.

Another considered medical image application is lossless compression. The objective has been to add the task of image segmentation to that of image compression such that the image regions can be transmitted separately, depending on the region of interest for diagnosis. The experiments performed on retinal color images show that our modeling, in which the MDL criterion selects the structure of the linear predictive models, outperforms publicly available image compressors such as the lossless version of JPEG 2000.

For time series modeling, the thesis presents an algorithm which allows detection of changes in time series signals. The algorithm is based on one of the most recent implementations of the MDL principle, the sequentially normalized maximum likelihood (SNML) models.

This thesis produces contributions in the form of new methods and algorithms, where the simplicity of information theoretic principles are combined with a rather complex and problem dependent modeling formulation, resulting in both heuristically motivated and principled algorithmic solutions.

Preface

The work presented in this thesis has mainly been carried out at the Department of Signal Processing, Tampere University of Technology, and a minor part has been carried out at the Department of Biomedical Engineering and Computational Science, Helsinki University of Technology.

First of all, I would like to express my sincerest gratitude to my supervisor Professor Ioan Tabus for providing me the opportunity to work in his group and for providing me such an interesting research topic. We have had numerous discussions during the years, and he has always been available whenever needed. His trust, guidance and support have been essential for this thesis. Emeritus Professor Jaakko Astola has provided his supervision, advice and support, which are greatly acknowledged. I would also like to thank Professor Moncef Gabbouj for his support and accepting me to work under his Big Data project. The former head of the Department of Signal Processing and the current Vice Dean for Research, Professor Ari Visa, is acknowledged for creating such a vibrant research environment. I would also like to thank Professor Jukka Heikkonen for his guidance during my early career at the Helsinki University of Technology, and for all the advice he has given.

The co-authors Dr. Andrea Hategan, Dr. Ionut Schiopu, and M.Sc.(Tech.) Pekka Astola are acknowledged for their proficient cooperation. I would also like to thank my co-author, Clinical Associate Professor M.D. Edmond Sabo, for introducing me to the world of histopathology and providing important feedback. M.D. Satu Haikonen is acknowledged for discussions on retinal images and their importance. I would also like to thank my roommates and all my colleagues during the years.

I have had the privilege to discuss in person with Emeritus Professor Jorma Rissanen. All those discussions have been very inspiring and rewarding.

I thank Pirkko Ruotsalainen, Virve Larmila, and Noora Rotola-Pukkila for their invaluable help on practical matters. Also, Elina Orava is acknowledged for her help in issues related to doctoral studies.

The pre-examiners Dr. Daniel Nicorici and Dr. Radu Bilcu are acknowledged for carefully evaluating the thesis and suggesting some minor changes that considerably

improved the quality of the thesis. In addition, I would like to thank Dr. Daniel Nicorici and Professor Cristian Perra for agreeing to be my opponents.

I would like to express my gratitude to Liisa Lund for her consultancy in language-related matters.

The thesis was financially supported by the Academy of Finland (under the grant 213462, Finnish Centre of Excellence Program 2006-2011), by the Doctoral Programme in Information Science and Engineering (TISE), by a scholarship from the Nokia Foundation, and by a grant from the Faculty of Computing and Electrical Engineering. Their support is greatly acknowledged.

Finally, I would like to thank my family. My parents Jukka and Mirja have always supported and encouraged me to reach my goals. They have not counted hours or driving kilometers to provide their help whenever needed. My brother Janne and sister Jonna have been invaluable by sharing their time with our family. Furthermore, my whole extended family is acknowledged for supporting our family. Above all, this work would have never been completed without my understanding and caring husband. Toivo, thank you for all the shared adventures in air, ground and water, and let there be many adventures ahead. Furthermore, I would like to thank you for being such a good father to our sons. Special thanks go to Veikko and Oiva, who were born during these years. You have shown what the most important things in the world are.

Tampere, May 2018 Jenni Hukkanen

Contents

Acronyms

AIC	Akaike's information criterion
AR	Autoregressive
ARMA	Autoregressive–moving-average
BIC	Bayesian information criterion
CAD	Computer assisted diagnosis
CALIC	A context-based, adaptive, lossless image codec
CERV	Crack-edge-region-value
CV	Cross-validation
DRIVE	Digital retinal images for vessel extraction
H&E	Hematoxylin and eosin
HT	Hough transform
JPEG-LS	Lossless compression standard
JPEG 2000	Compression standard created by Joint Photographic Experts Group committee in 2000
LCIC	Lossless color image compression algorithm
LOCO-I	Low complexity lossless compression for images
LOO	Leave-one-out
MDL	Minimum description length
NML	Normalized maximum likelihood
RCT	Reversible color transform
SNEF	Segmentation of nuclei by ellipse fitting
SNML	Sequentially normalized maximum likelihood
SOM	Self-organizing map

List of Publications

I J. Hukkanen, A. Hategan, E. Sabo, and I. Tabus, "Segmentation of cell nuclei from histological images by ellipse fitting," in *Proceedings of the 18th European Signal Processing Conference (EUSIPCO-2010)*, Aalborg, Denmark, August 2010, pp. 1219 – 1223.

II J. Hukkanen, E. Sabo, and I. Tabus, "Representing clumps of cell nuclei as unions of elliptic shapes by using the MDL principle," in *Proceedings of the 19th European Signal Processing Conference (EUSIPCO-2011)*, Barcelona, Spain, August 2011, pp. 1010 – 1014.

III J. Hukkanen, E. Sabo, and I. Tabus, "MDL based structure selection of union of ellipse models for scaled and smoothed histological images," *Advances in Intelligent Control Systems and Computer Science*, Springer Berlin Heidelberg, pp. 77 – 89, 2013.

IV J. Hulkkonen* and J. Heikkonen, "A minimum description length principle based method for signal change detection in machine condition monitoring," in *Proceedings of the 19th International Conference on Pattern Recognition*, Tampa, Florida, December 2008, pp. 1 – 4.

V I. Tabus, J. Hukkanen, and I. Schiopu, "Two-phase compression of histological images with MDL ranking of segmentation images," in *Proceedings of the 19th International Conference on Control Systems and Computer Science*, Bucharest, Romania, May 2013, pp. 331 – 338.

VI J. Hukkanen, P. Astola, and I. Tabus, "Lossless compression of regions-of-interest from retinal images," in *Proceedings of the 5th European Workshop on Visual Information Processing (EUVIP2014)*, Paris, France, December 2014, pp. 1 – 6.

* The former last name of Jenni Hukkanen was Hulkkonen.

1 Introduction

1.1 Motivation of the thesis

Advancing methods for medical image analysis and compression is becoming more and more important, since medical images are becoming more available in clinical practice due to the availability of high quality imaging devices. As imaging systems are improving, the sizes of medical images are also growing because of higher spatial resolution and a higher number of bits per pixel. Also, the number of taken images is increasing, as image acquisition systems are getting cheaper and a greater amount of medical images are routinely taken. The workload of medical doctors has also been increased since they have to analyze, handle and store an increasing number of images. These facts have led to a situation in which more and more medical doctors' time is spent with medical image analysis tasks. As an ideal goal, many images could be easily analyzed automatically by computer programs, and only those images, or part thereof, that are difficult to diagnose could be delivered to medical doctors for assessment. Therefore, there is a need for automatic and semiautomatic image processing and analysis methods that would allow medical doctors to concentrate on diagnostically difficult cases and to shift their focus towards diagnostically important parts of the images.

Two important functionalities for efficient digital image analysis and processing are segmentation and compression [1, 2]. The aim of segmentation is to split the image into regions for simplifying and representing it in a form useful for the following image analysis stage, e.g. detection of the objects' shape. It is very important that image segmentation is highly accurate, since the failures made in segmentation can not be later recovered. The obtained segmentation can also be further applied to the task of compression. Compression allows the image to be stored and transmitted using a smaller amount of bits than the original image. In lossless compression, all the information from the original image is preserved, and from the compressed image one can fully recover the original image. In medical images, lossless compression is extremely important since no loss of information is allowed on diagnostically important regions. Therefore, combining compression with segmentation so that lossless encoding could be applied only to the regions-of-interest can save storage space and transmission time.

Manual segmentation of diagnostically important patterns from medical images is often a long and time-consuming task. Several automatic and semiautomatic image segmentation algorithms have been proposed, see for instance [1]. However, they are not always directly applicable to medical images, since segmentation often requires application specific knowledge. Some of the main difficulties for segmentation algorithms are due to the existence of texture, occlusions and corrupting noise in the image content. The two unwanted situations for a segmentation result are oversegmentation and undersegmentation. In oversegmentation, the image is split into too many regions, while in undersegmentation, the regions are too large and one single region may expand over several distinct objects.

Whenever, the objects in the image are overlapping or touching, forming clumps, the segmentation may not provide the correct object separation. There might not be any gradients or intensity variations between the objects which would guide the traditional image segmentation algorithms to segment the individual objects from the clump. Therefore, some prior assumptions about the shapes of objects are necessary. A wide variety of objects can be modeled well by convex and elliptical shape priors. The most used approaches can be divided into two classes. The first category of approaches is splitting a clump into smaller non-overlapping pieces. They are mostly used on a binary image obtained by some segmentation algorithm. The approaches include shape-based watershed [3], and concavity analysis based methods, such as [4]. The main disadvantage of these approaches is that the binary image may already contain some distortions.

The second category of clump splitting approaches contains a wide variety of model-based approaches, which aim to detect from the clump several objects with some predefined shapes. The advantage of these models is that they allow objects to be overlapping in the image representations. In the fields of computer vision and pattern recognition, different detection approaches for elliptical objects from images have been widely studied. Most of them also work on binary images, such as binary edge images. The approaches include, for instance, Hough transform [5, 6, 7]. The method has some drawbacks, namely it is computationally inefficient [8, 9, 10]. In addition, the shapes of the wanted objects need to be defined very precisely [9]. The state of the art method for detecting multiple ellipses concentrates on efficiently grouping the edge pixels into segments of possible arcs of the ellipses [11]. Also, combinations of both applying concavity analysis and fitting ellipses to the smoothed contour segments of a clump have been seen in practical applications [12]. Therefore, there is a need for algorithms that efficiently detect the locations of several ellipse-resembling objects and that evaluate the results based on the original image, not on a binary image resulting from preliminary segmentation or on a binary edge image, as in case of many approaches.

In image compression algorithms, the two key components are modeling and coding [2]. The aim of modeling is to try to predict the values of image pixels

close to the actual ones. In the coding stage, the differences between the predicted and true values are encoded. These differences are also called residuals. Since pixel values are often spatially correlated, many predictive image compression algorithms, such as CALIC [13] and LOCO-I [14], utilize prediction based on the values of the pixels in a causal template, also called a prediction context. A causal template consists of already processed close-by pixels. The size of the templates and their shapes varies for different compression algorithms. The encoding contexts are sometimes different from the prediction contexts, and they are used for collecting the encoding distribution for the prediction residuals. The encoding contexts are hence used to remove the remaining correlations after the prediction stage, by grouping similar neighborhoods to be encoded separately. The reason for this is that smooth and fast-changing image areas most likely have different distributions of residuals, and for efficient encoding of residuals one needs to use distributions as close as possible to the true ones. In sparse predictive lossless image compression, the causal template elements are selected by sparse prediction design methods, making use of the algorithms for sparse modeling [15].

One important aspect regarding accurate image segmentation and compression is the evaluation, comparison and ranking of different solutions. Segmentations can be compared against ground-truth segmentations, if available, and in the case of image compression, the more the image can be compressed, the better the compression algorithm is performing. However, how do we know how the developed algorithm, method or model will work on similar data that we have not yet tested on? Which method or model is describing best the phenomenon that we are currently studying? That is a task which calls for model selection. The general problems regarding model selection are over- and underfitting. In underfitting, the selected models are too simple to describe all the necessary aspects of the phenomena, and better fitting models exist. Whereas, in case of overfitting, the models are too complex: they fit the data well, but they are too detailed, which causes that their ability to generalize to future data is reduced. Several approaches for model selection have been proposed, which include non-parametric approaches such as cross-validation (CV) [16] and bootstrapping [17], and parametric model selection approaches such as Akaike's information criterion (AIC) [18], Bayesian information criterion (BIC) [19], and the minimum description length (MDL) principle [20, 21]. In this thesis, model selection is utilized in three different modeling scenarios. First, model selection has been used to select between image interpretations, i.e. the number of ellipses, their locations, and shapes (Publications I, II, and III); second, model selection has selected the structure of the linear predictive models (Publication VI); and third, the number of previous time step used in autoregressive (AR) models is selected by a model selection method (Publication IV).

The MDL principle provides an efficient framework for model selection. It is inspired by Kolmogorov complexity [22]. The idea of the MDL principle is to equate learning with finding regularities in data, since any regularity can be

used to compress that data. Therefore, MDL aims to find in a set of models that model structure which gives the lowest total codelength for both the data and the model. Over the years, several methodologies have been developed, following the ideology expressed by the MDL principle. Two-part coding [20] is the earliest implementation, and provides the simplest and most intuitive embodiment of the MDL principle, being the only implementable approach in some specific applications. The second main MDL approach is the normalized maximum likelihood (NML) model [21, 23], which departed from the separate two-part coding, by using in coding a single normalized distribution, which is a very elegant approach, but is rather complex to implement. A more recent MDL method is based on the sequentially normalized maximum likelihood (SNML) models [24, 25], which are especially designed for time series data, being introduced to overcome some of the problems encountered with the NML models, especially the implementation complexity issue. In the field of image segmentation, MDL was first introduced by Leclerc [26]. Kanungo [27] proposed a two-part coding- and region-merging-based image segmentation algorithm for multilayer images such as color images. A similar approach was also taken by Luo [28], although Luo developed the approach further by adding smoothing to obtain segmentations at multiple scales, and left the selection of the correct scale as a task for the user of the algorithm.

1.2 Objectives of the thesis

The main objective of this thesis is to develop model selection techniques for medical image segmentation and compression. One main application considered in this thesis is segmentation and clump splitting of cell nuclei in histological images. Histological images are images of thin tissue samples, in which the wanted structures are highlighted by a specific staining. In hematoxylin and eosin (H&E) stained histological images, the cell nuclei are shown with a bluish color and their shapes can be approximated by ellipses. Histological images prove to be challenging for segmentation and clump splitting algorithms. The reason is that the intensity within cell nuclei may vary. In addition, the background can be very complex and segmenting image into regions of cell nuclei and background can be difficult.

The individual objectives of the thesis are summarized as follows:

- to develop segmentation and clump splitting algorithms for cell nuclei segmentation in histological images;
- to improve the performance of the heuristic segmentation algorithms by adding an information-theory-inspired criterion for ranking different clump interpretations;

- to show by experimental verification that the proposed MDL-based criterion is selecting the interpretation that is among the ones closest to the ground truth interpretation;
- to add a segmentation stage into linear predictive lossless image compression algorithms and to analyze their compression performances on histological images;
- to propose a lossless medical image compression algorithm in which the structure of the linear predictive model is selected by an MDL-inspired criterion and;
- to develop a signal change detection algorithm in which the MDL-based estimate of the signal complexity is applied to detect changes in time series signals.

1.3 Author's contributions

The research work which led to the publications presented in this thesis was mainly conducted at the Department of Signal Processing, Tampere University of Technology, and the work was supervised by Prof. Ioan Tabus. The work for Publication IV was performed at the Department of Biomedical Engineering and Computational Science, Helsinki University of Technology, and supervised by Prof. Jukka Heikkonen. The author of the thesis is the first author in Publications I, II, III, IV, and VI, and the second author in Publication V. Next, a brief description of the contributions to each publication is given.

Publication I: The publication proposes an ellipse fitting based cell nuclei segmentation algorithm for histological images. The author of this thesis has combined ideas of the first and second author and implemented them as the proposed algorithm. The writing of the publication was done in collaboration with the fourth author.

Publication II: The publication proposes an MDL-based criterion for ranking of different clump interpretations. Compared to existing MDL-based criteria for image segmentation, the proposed criterion uses a codelength, which is obtained by encoding on a real computer program, and hence asymptotic approximations of codelength can be avoided. Additionally, the criterion is suitable for solving applications with clumps of overlapping nuclei. The final form of the criterion is the result of the collaboration of the author of the thesis and the third author. The author of the thesis is responsible for the implementation of the criterion. The analysis of the results and the writing of the publication was done in collaboration with the third author.

Publication III: The publication applies the criterion proposed in Publication II and shows that the criterion is applicable to select the interpretation that is

among the ones closest to the ground truth interpretations. The author of this thesis has implemented the experiments. The writing of the publication was done in collaboration with the third author.

Publication IV: The publication applies the sequentially normalized maximum likelihood (SNML) criterion to time series modeling. The publication proposes an algorithm for detection of changes in time series signals. The author of this thesis has implemented the algorithm and is responsible for the experiments described in the publication. The writing of the publication was done in collaboration with the second author.

Publication V: The publication proposes four different lossless image compression algorithms for gray level histological images. The author of this thesis has contributed to the publication by experimenting with the mean shift segmentation algorithm. In addition, the author of this thesis has participated in the discussions of the proposed image compression algorithms. The writing of the publication was done in collaboration with the authors of the publication.

Publication VI: The publication proposes a lossless image compression algorithm for retinal images. The algorithm selects the structure of the linear predictive model by an MDL-inspired approach. In addition, the algorithm allows the regions-of-interest to be transmitted independently, once the algorithm has transmitted the contours of the segmentation regions first. The author of this thesis has influenced to the development of the algorithm. The writing of the publication was done in collaboration with the authors of the publication.

1.4 Structure of the thesis

The compendium part of the thesis gives first the background on the topics treated in the collection of six original publications. The rest of the introductory part is structured as follows. Chapter 2 gives an introduction to a few elementary building blocks used in image segmentation applications. Chapter 3 presents the new approaches to segmentation of cell nuclei from histological images. Chapter 4 concentrates on information-theory-inspired approaches for segmentation and model selection. First, the chapter gives a brief introduction to model selection. Then, we review few concepts from information theory and data compression, necessary for implementing the MDL principle to solve the problem of model selection. The model selection tool used in this thesis is the minimum description length principle, which is briefly introduced in Chapter 4, together with the sequentially normalized maximum likelihood (SNML), which is a modern embodiment of the MDL principle. Chapter 5 discusses lossless image compression algorithms. Finally, Chapter 6 summarizes the thesis and gives some research ideas for future development.

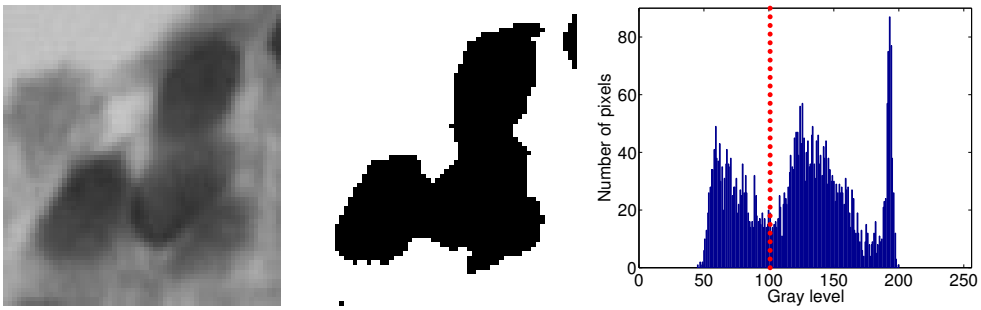
2 Image segmentation building blocks for the proposed algorithms

Segmentation is a process that splits images into several parts, often called regions [1]. These regions can be for instance foreground and background, or object(s) and background. The main goal of the segmentation is to simplify and represent images in a form that is easier to analyze. For instance, detection of objects, and their orientation, size, or their relative positions may be wanted properties for later analysis. Therefore, the success of the segmentation process is a precondition for the success of the whole signal analysis process as the failures made in segmentation cannot be recovered later.

Several digital image segmentation algorithms that are used alone or aggregated in more complex methods are presented in the following. The segmentation regions are usually characterized by having similar properties, e.g. intensity, color, or texture, within the region, and by having different properties compared to the background and other objects. Another way to define segmentation regions is to locate their borders where the gradient has high values.

The main error situations for automatic segmentation algorithms are oversegmentation and undersegmentation. Intensity variations within one object may split it into more than one region, causing oversegmentation. The other error case, undersegmentation, is often caused by overlapping and touching objects. They produce clumps or clusters which are difficult to solve by ordinary segmentation algorithms.

The main goal of this thesis is to develop new algorithms for medical image segmentation. In this chapter, we will concentrate on some basic preliminaries for segmentation algorithms which include thresholding, gradient magnitude image, estimating the locations of edge pixels, and some specific image segmentation algorithms, such as mean shift segmentation.



(a) (b) (c)

Figure 2.1: Thresholding H&E stained tissue image. (a) Gray scale image. (b) Thresholded binary image. (c) Histogram of the gray scale image with the threshold (in red) being 100.

2.1 Image thresholding

Thresholding is a simple image segmentation approach. It converts a gray scale intensity image into a binary image by comparing the pixel intensities to a threshold value. The pixel values which are less than the threshold are marked as objects and the background pixels are the remaining pixels. The threshold value is typically determined based on the histogram of the image pixel intensities. The main advantage of thresholding is its speed: the algorithm produces preliminary segmentation results fast and the computational burden is low. Although thresholding is rarely enough to produce the final segmentation, it often produces good estimates for further processing and serves as a starting point for more advanced segmentation algorithms. Next, two thresholding algorithms used in Publications I, II and III are presented. Detailed overviews of image thresholding algorithms can be found, for instance, in [29, 30].

One popular thresholding algorithm is Otsu's method [31]. It assumes that the histogram of pixel intensities is bimodal. This means that Otsu's method assumes that there are two classes in the image: the foreground and the background. Otsu's method aims to find the threshold T by minimizing the intra-class variance of the two classes, which is the weighted sum of variances of the two classes, presented in [31] as

$$\sigma_w^2 = w_1\sigma_1^2 + w_2\sigma_2^2, \tag{2.1}$$

where the class probabilities $w_1 = \sum_{i=1}^{T} p_i$ and $w_2 = \sum_{i=T+1}^{L} p_i$ are calculated from the histogram of intensities $p_i = n_i/N$, where n_i is the number of pixels at the intensity level i, the total number of pixels is $N = \sum_i^L n_i$, and L is the number of intensity levels. The corresponding class variances are given as $\sigma_1^2 = \sum_{i=1}^{T}(i-\mu_1)^2 p_i/w_1$ and $\sigma_2^2 = \sum_{i=T+1}^{L}(i-\mu_2)^2 p_i/w_2$, where the mean values of the classes are $\mu_1 = \sum_{i=1}^{T} i p_i/w_1$ and $\mu_2 = \sum_{i=T+1}^{L} i p_i/w_2$. The optimal threshold value is obtained by an exhaustive search.

The other used thresholding algorithm is the dual thresholding method [32]. Compared to Otsu's method, dual thresholding aims to find two thresholds, denoted as T_1 and T_2. The idea of the two thresholds stems from images having three classes. For instance, in histological H&E stained tissue images, the three classes consist of nuclei, cytoplasm and background. The dual thresholding algorithm is as follows. First, the image histogram is divided into three parts, C_1, C_2 and C_3, such that the thresholds T_1 and T_2 divide the histogram into three equal sized regions: $T_1 = L/3$ and $T_2 = 2L/3$, where L denotes the number of gray levels. The thresholds T_1 and T_2 are updated as $T_1 = (\mu_1 + \mu_2)/2$ and $T_2 = (\mu_2 + \mu_3)/2$, where μ_1, μ_2 and μ_3 are the average intensities of the classes. The loop is repeated until the values T_1 and T_2 converge, or the maximum number of iterations is reached.

An example of thresholding on a histological tissue image is shown in Figure 2.1. The original gray scale intensity image is presented in Figure 2.1(a) and the thresholded binary image is shown in Figure 2.1(b). The histogram of the gray scale image with a threshold is shown in Figure 2.1(c). The threshold value is obtained by using the dual thresholding algorithm, and due to visualization purposes, only the lower threshold value, T1, is applied and shown in Figures 2.1 (b) and (c).

2.2 Gradient magnitude image and location estimation for edge pixels

An edge is a sharp, local change in image intensity. Edges are important in image segmentation, since it is often desirable that the borders of the segmentation regions are placed into fast-changing image intensity locations [1]. Edges can be detected using a gradient magnitude image, which represents local contrast in an image such that high values correspond to sharp edges and low values to uniform areas. In Figure 2.2(b), we have shown a gradient magnitude image in which light gray corresponds to high gradient values and dark colors to constant areas. We have obtained the gradient magnitude images using Sobel operators [1]. The operators are 3×3 kernels which are convolved with the original image I such that the resulting approximations of the gradients in horizontal and vertical directions are

$$\mathbf{g}_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I \qquad \text{and} \qquad \mathbf{g}_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I, \tag{2.2}$$

where $*$ denotes the convolution operation. Then, the gradient magnitude image can be computed as in [1]:

$$\mathbf{G}(i,j) = \sqrt{\mathbf{g}_x(i,j)^2 + \mathbf{g}_y(i,j)^2}, \tag{2.3}$$

(a) **(b)** **(c)**

Figure 2.2: Gradient magnitude image and thresholded gradient magnitude image. (a) Original gray scale intensity image. (b) Gradient magnitude image. (c) Thresholded gradient magnitude image.

where (i, j) denotes the location of a pixel in the image. Other possible filter kernels for gradient magnitude exist, e.g. the Prewitt operator [1].

A gradient magnitude image can be used as a preliminary stage in an image segmentation algorithm; for instance, the watershed segmentation algorithm [33] is often performed on a gradient magnitude image instead of the original image. The other approach is to threshold the gradient magnitude image, which gives estimates for the locations of edge pixels. One of the main difficulties for edge pixel estimation is caused by noise. Smoothing can be used to alleviate the problems, but the smoothing may also distort important edges. In addition, edge pixel sets can rarely be used to directly produce segmentations, since there are often discontinuities in the edge pixel sets so that they do not enclose closed regions. We will discuss the use of an edge image on elliptical object detection later in Section 3.3.3.

In Figure 2.2, we have shown a gradient magnitude image and its thresholding by Otsu's method. A gray scale intensity image and its gradient magnitude image are presented in Figures 2.2(a) and (b), respectively. The thresholded gradient magnitude image for which the threshold value is obtained by Otsu's method is shown in Figure 2.2(c).

2.3 Introduction to three specific image segmentation algorithms

Next, we will describe segmentation algorithms that are relevant to this thesis: region growing, watershed, and mean shift clustering based segmentation.

2.3.1 Region growing

Region growing [34, 35, 36] aims to divide an image I into homogeneous regions $R_1, \dots R_m$ by starting with small regions and merging neighboring regions based on some criterion. The regions are merged until no neighboring regions can be merged.

A widely used criterion is Fisher's test [37]. The squared Fisher distance between two adjacent regions R_1 and R_2 with respective sizes, sample means and sample variances $n_1, n_2,\ \hat{\mu}_1,\ \hat{\mu}_2,\ \hat{\sigma}_1^2,\ \hat{\sigma}_2^2$ is presented in [36] as

$$\frac{(n_1 + n_2)(\hat{\mu}_1 - \hat{\mu}_2)^2}{n_1\hat{\sigma}_1^2 + n_2\hat{\sigma}_2^2}. \tag{2.4}$$

If the value is below a certain threshold, the regions are merged.

In [36], it has been discussed that region growing algorithms rarely converge to the global minimum of a cost function and that the resulting boundaries may be noisy. In addition, the other problem with Fisher's test is that it merges regions having equal means, but different variances [36]. Some of the problems can be alleviated by starting the region growing from reasonably sized regions, or by using more sophisticated measures. The minimum description length (MDL) principle based merging measures for region growing will be discussed in Chapter 4.

2.3.2 Watershed

One version of region growing is watershed [33]. In watershed, an intensity image is commonly interpreted as a landscape, where the height of the landscape corresponds to the intensity value. The segmentation regions are then the drainage regions on the landscape. Therefore, the regions are obtained by placing water sources into regional minima of the landscape, which form catchment basins, and allowing water to flood level-by-level from the catchment basins. A watershed, or boundary between two regions, is placed at the meeting points of two different catchment basins.

Instead of using an intensity image as a landscape, more often it is preferred to use a transformed image, such as gradient magnitude image. The reason is that dark object regions are rarely separated from background by light ridges but more likely by high changes in intensity. In gradient magnitude images, high values of gradient magnitude correspond to sharp edges and low values to uniform areas. Hence, watershed places the boundaries of the regions to the highest points of the ridges, which corresponds to the fastest change in intensity. The gradient magnitude image also allows watershed to be applied to color images.

The original version of watershed is sensitive to noise and often produces many small regions, as the number of regions equals the number of water sources, or seeds. Approaches to improve the results include smoothing (as a pre-processing)

and merging of the regions based on rules (as a post-processing). Marker-based watershed [3] can remedy the issue by estimating markers that belong to the same region. Despite all the efforts done to improve the original watershed, it is not able to split clusters of touching objects if there is no intensity variation between the objects. There exist some efforts to split clumps of objects by applying watershed twice: first, the ordinary watershed is applied, and on the second watershed round, the watershed is applied to the complement of the distance transform. Other approaches to add prior information to improve the watershed results include e.g. [38, 39]. The clump splitting methods are discussed in more detail later in Section 3.2.

2.3.3 Segmentation based on mean shift clustering

Mean shift is a non-parametric clustering approach originally presented in 1975 by Fukunaga et al. [40]. The idea behind mean shift clustering is that it efficiently finds the modes of high dimensional data distributions without explicitly estimating the density functions. An estimate for the data density function is obtained by kernel density estimation or by the Parzen window technique [41, 42, 43], which give a smoothed estimate of the data density by convolving the data samples with a fixed kernel. The modes of the density function are found from the zeros of the gradient of the density function, and the mean shift vectors point to the direction of the maximum increase in the density. Therefore, the data samples are clustered based on the modes of the estimated density function, such that a cluster consists of data samples that have their trajectory of mean shift vector locations converging to the same mode in the estimated density function. In mean shift image segmentation [44], the color or spectral values are clustered jointly with the pixel locations, and the segmentation regions consist of corresponding mean shift clusters.

The advantages of mean shift clustering and segmentation is that it does not assume any underlying data distributions. The clustering is scaled with a single parameter, the width of the kernel window, which is also known as the bandwidth. Usually, small window widths correspond to many small clusters, and large window widths give few large clusters. In mean shift image segmentation, there are often two parameters: range- and spatial-bandwidths. They allow the use of different scales for pixel color and locations in the kernel function. One of the main challenges in applying mean shift clustering is that the width of the window needs to be selected so that it is proper for the current application. An algorithm for data-driven window width selection is proposed in [45].

In this thesis, we have applied mean shift segmentation in Publication V, which proposes several two-phase lossless image compression algorithms for histological images. The algorithms encode both the segmentation and the values of the original image. The goal of the two-phase compression algorithms is that they could be used to rank different image segmentations. In addition, segmentations

might help in the encoding of the images. In the experiments presented in Publication V, we obtained several mean shift segmentations by varying the bandwidth parameters. At the beginning, the parameters were coarse and the scale of the parameters large, so that we had several segmentation images which ranged from highly oversegmented images to highly undersegmented images. Then, the ability of the two-phase compression algorithms to rank the segmentations based on the total codelengths was studied. For more discussion on Publication V, see Section 5.4.

3 Segmentation of cell nuclei from histological images

In the previous chapter, we gave an introduction to image segmentation approaches relevant to this thesis. However, the ordinary image segmentation algorithms are not usually enough when there are overlapping and occluding objects in the image. These objects need special attention, since there might not be any gradient between the objects which would guide segmentation algorithms to separate them into individual ones. A good example is the segmentation of cell nuclei from histological images. Cell nuclei are often overlapping in the acquired 2D images, so that ordinary segmentation algorithms can only give an estimate for the contour of the cell nuclei clump. Cell nuclei can often be modeled closely enough by ellipses, and therefore, we will concentrate on approaches for separating and splitting clumps of ellipse-resembling objects.

The structure of this chapter is as follows. First, we give an introduction to segmentation of H&E stained histological images. Then, we present three general approaches to the separation of overlapping and touching ellipse-resembling objects. After that, we concentrate on model-based approaches, and especially approaches that are based on ellipses. First, we present two parameterizations of ellipses that are needed in this thesis. Then, we review approaches for fitting an ellipse to image pixel coordinates. After that, we describe the difficulties of fitting several ellipses to binary edge image or to the contour of a clump. Finally, we present our SNEF algorithm, proposed in Publication I. The algorithm fits ellipses to a specific edge image, obtained by combining intensity and gradient information. The algorithm proposes several candidate ellipses, out of which the ellipses for the final representation of the clump are selected by the proposed goodness-of-fit criterion.

3.1 An introduction to segmentation of H&E stained histological images

One important application field for clump splitting algorithms is provided by histological images [46]. Histological images are images of thin tissue samples of

biopsies. The tissue samples are processed and fixed onto glass slides. After that, the glass slides are screened to study signs, grades, and prognoses of diseases. The preparation process of the histological slides aims to preserve the tissue architecture, so that they provide a comprehensive view of the tissue for disease grading. Pathology diagnoses are currently given by pathologists after careful evaluation of histological slides. However, the educated opinion of the pathologist for diagnosis is subjective, since some amount of inter-observer variations between diagnoses have been reported, e.g. [47]. In addition, due to the vast amount of histological images that a pathologist screens daily, the workload is enormous and most of it is spent on obviously benign areas [46]. Hence, there is a need for computer-assisted diagnosis (CAD), in which the aim is not only to reduce the effects of subjective opinion, but also to allow the pathologist to focus on diagnostically difficult cases. Furthermore, knowledge gathered from quantitative analysis of histological images can be used to understand the biological mechanisms of disease processes.

Diseases in histological images are characterized mainly by cell nuclei [46]. Some important features of cell nuclei for diagnosis include e.g. size, shape, orientation, eccentricity, intensity, texture, and chromatin-specific features. The wanted structures of tissue can be emphasized in an image by using a specific staining. In histological images, a commonly used staining is hematoxylin and eosin (H&E), which colors cell nuclei blueish and cytoplasm and other remaining tissue parts with shades of pink. An H&E stained histological image is shown in Figure 3.1.

Difficulties for the automatic cell nuclei segmentation algorithms are caused by the complex nature of histological images. The internal variations within nuclei can be greater than those between the individual ones. In addition, the background consisting of cytoplasm and other tissue parts is neither constant nor easy to segment. Naturally, basic thresholding and finding the correct threshold value is difficult in these kind of images. On the other hand, more refined segmentation algorithms can be time consuming and do not guarantee proper segmentation results either. Some approaches to cell nuclei segmentation have been proposed, which include median filtering and thresholding [48], adaptive thresholding and morphological operations [49], and Bayesian classifier and template matching by four elliptical templates with different major and minor axes [50].

Some algorithms have been developed especially to separate clumps of cell nuclei from histological images. The reason for the clumps occurring in histological images stems from the thickness of sample sections. The 3D tissue samples are sliced into thin sections. However, the thickness is not small enough so that we could observe only well-separated cell nuclei in the acquired 2D images. Approaches to solving the problem of overlapping or touching cell nuclei clumps in histological images include e.g. a concave point based approach [51].

A number of nuclei clump splitting algorithms have been proposed to cytological images, which are images closely related to histological images. Cytological images

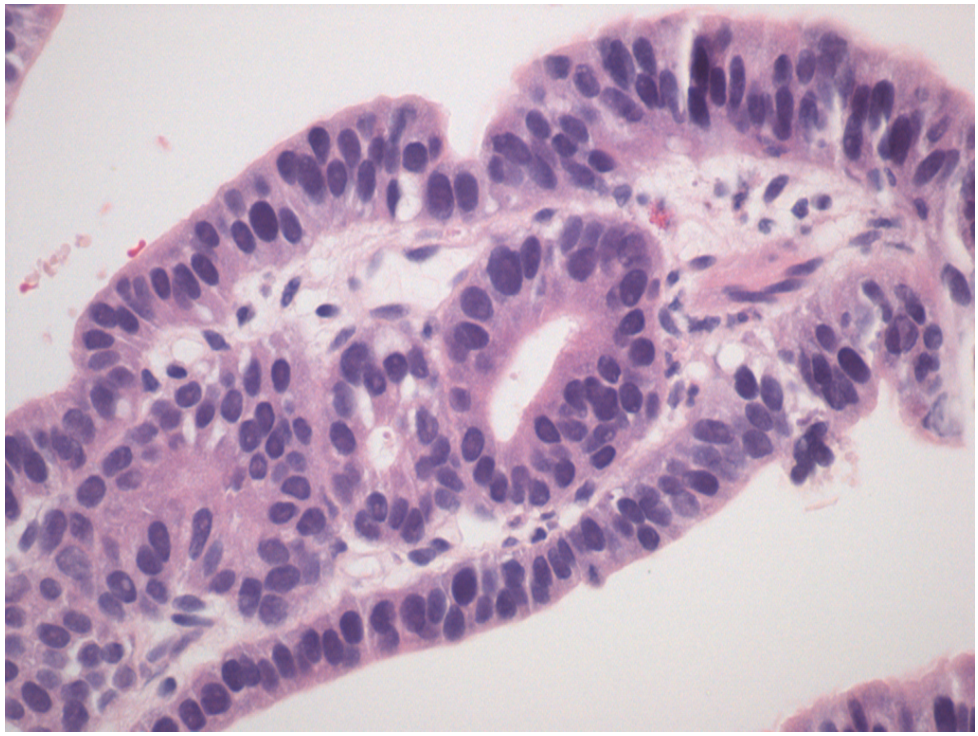


Figure 3.1: An H&E stained histological image, where cell nuclei are bluish and their shape is close to ellipses. Some cell nuclei are touching each other and forming clumps of nuclei.

are taken from less invasive biopsies and contain samples of free cells or tissue fragments, such as a cervical Pap smear [52]. Cytological images are often easier to segment than histological images, as cytological images do not usually preserve tissue architecture and lack more complicated structures such as glands [46]. The clump splitting algorithms for cytological images include model-based approaches such as deformable templates [9], active shape models [53], and a watershed-based approach [54].

We are interested in model-based approaches for cell nuclei segmentation from histological images. We are especially interested in representing cell nuclei by ellipses such that one ellipse represents one nucleus. The motivation for elliptical shapes in cell nuclei segmentation and clump splitting does not only stem from the convex and ellipse-resembling shape of nuclei, but also from the desired features of nuclei used in histopathological image analysis. The wanted nuclei features include especially the lengths of major and minor axes, eccentricity, orientation, and elliptical deviation [46], and those can be easily estimated from ellipses.

Next, we will give a general overview to separation of overlapping and touching ellipse-resembling objects. Then, we will concentrate on model-based and especially ellipse-fitting-based approaches.

3.2 General overview to separation of overlapping and touching objects, resembling ellipses

One important aspect of segmentation and object detection is splitting clumps of objects. The clumps of objects are formed by objects such that the objects are overlapping or touching each other so that many of the segmentation algorithms as such are not able to separate them into individual objects. Naturally, one of the most important prior information for solving the problem of clustered objects is the shape of the objects. Here, we concentrate on objects having a convex shape, which include e.g. roundish and ellipse-resembling objects.

The splitting and separation algorithms for clumps of convex objects can be divided into watershed-based, model-based, and methods based on concavities. Many of the splitting approaches, for instance most of the watershed- and concavity-based approaches, are working over binary images obtained by a segmentation, or an edge detection algorithm. The drawback of these kinds of two-phase approaches is that not all the information from the original image is used when the splitting decision is made.

Shape-based watershed segmentation separates clustered objects based on roundness. In the approach, the watershed algorithm is applied twice. First, the original image is segmented by the ordinary watershed algorithm. Then, the binary segmentation image is transformed into a distance image, where for each foreground pixel the distance to the nearest background pixel is shown. Finally, the watershed segmentation is applied to the distance image. Due to distance transform, the approach is efficient with roughly circular objects [55, 56]. However, a large contact zone of objects, resulting from a large number of touching objects or objects being very close, may cause the clump splitting to fail, as noted in [8].

Model-based approaches contain a wide range of approaches. The similarity of the approaches is that they have a parametric model that is fit to the original image, edge image, or to the smoothed contour of a clump. The advantage of using models in clump splitting is that the model can be defined to take into account the values of the original image, not just fitting to the binary segmentation results. In addition, models can be specified such that they allow overlapping regions in results, which might be a desired property with occluding objects. The main problem of the model-based approaches is computational complexity. Many times the proposed clump splitting algorithms are applicable only to a couple of objects within the clump, e.g. [57]. The problem can be alleviated by effective pre-processing that restricts the parameter space and proposes preliminary clump splitting results for further optimization. Therefore, defining a model for clump splitting is in general a compromise between the accuracy of the results and the execution time. In [12], the problem of clump splitting is solved by concavity analysis and ellipse fitting to the smoothed contour segments of the clump. The final representation for the clump is done by a rule-based selection of the ellipses. Since this thesis

concentrates especially on the clump splitting of ellipse-resembling objects, the ellipse-fitting-based algorithms are discussed in detail later in Section 3.3.

Methods based on concavities are intuitive approaches to the splitting of clumps of convex objects. The concavities on the contour of the segmented clump are potential starting points for the candidate splitting lines. Hence, the algorithms based on concavity analysis typically consist of two phases: finding the potential starting points, i.e. mostly concavities, and then finding the corresponding starting points to be linked together to form a splitting line. A robust rule-based approach for clump splitting that is strongly based on concavity analysis is introduced in [4]. In the algorithm, the concavity points of the clump are found first, after which several rules are applied to generate candidate split lines. Finally, the best split line is selected by a proposed measure of split. In [56], a concavity-based approach is used to separate touching grains. The concavities are found by the morphological skeleton calculated from the background of the thresholded image. The splitting lines are found by starting from the open lines of the skeleton and prolonging them according to the direction derived from the skeleton. The prolonged lines that get closer than a certain value are connected, such that a line between the respective starting points is drawn.

3.3 Model-based approaches to detect elliptical objects from images

Ellipse detection is one of the most fundamental tasks in pattern recognition and computer vision, and has hence gained a lot of attention [11, 58, 59]. An ellipse is the perspective projection of the circle, and has five independent parameters instead of the circle's three parameters, which makes it the more general shape and more often perceived. Ellipse detection algorithms have been applied to several applications, including grain detection, industry robot vision, and medical image applications.

Next, we will present two parameterizations of ellipses. Then, we will describe the fitting of an ellipse to 2D coordinate points and after that, an overview to detection of multiple ellipses from an image.

3.3.1 Two parameterizations of an ellipse

We will introduce two ellipse parameterizations used in this thesis. The first parameterization can be used in the fitting of an ellipse to pixel coordinates. Some approaches to fit ellipses into image coordinates are given in Section 3.3.2. The second parameterization describes the ellipse by using the locations of the center point, length of major and minor axes, and the angle between the x-axis and the major axis. We have used the second parameterization in Publications II and III to describe the region boundaries. Naturally, many other ellipse parameterizations

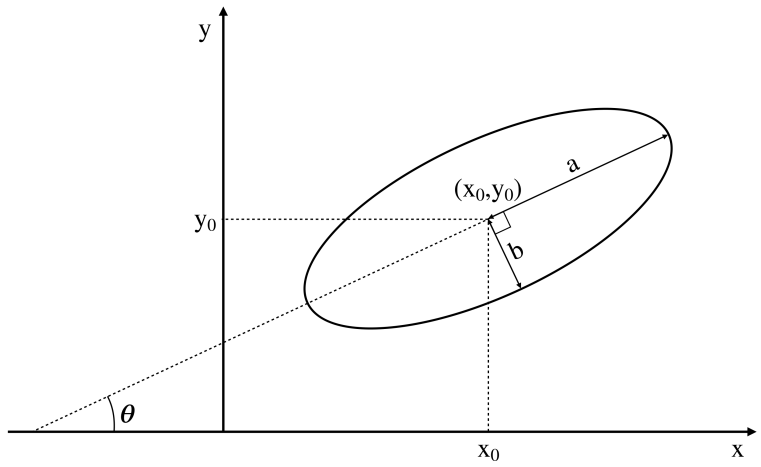


Figure 3.2: An ellipse with its five parameters: location of the center point (x_0, y_0), the lengths of the major and minor semi-axes a, b, respectively, and the rotation of the axes θ.

exist, e.g. [57, 60], and all of them can be transformed to these parameterizations. Next, the two parameterizations used in the thesis and their connecting transforms are presented.

An ellipse is a conic and it can be described by an implicit second-order polynomial in such a way as shown e.g. in [61]:

$$P(\mathbf{x}; \mathbf{a}) = Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \tag{3.1}$$

with an ellipse-specific constraint $B^2 - 4AC < 0$, where $\mathbf{a} = [A, B, C, D, E, F]^T$ are the ellipse parameters, and $\mathbf{x} = (x, y)$ gives the coordinates of the points lying on the ellipse.

Another way to describe an ellipse is to use center points (x_0, y_0), the length of the major and minor semi-axes, a and b, respectively, and the angle between the major axis and x-axis, θ. An ellipse with its five parameters are visualized in Figure 3.2. The ellipse equation is given as

$$\frac{x'^2}{a^2} + \frac{y'^2}{b^2} = 1, \tag{3.2}$$

where the coordinates x' and y' after translation and rotation are

$$\begin{aligned} x' &= (x - x_0)\cos\theta + (y - y_0)\sin\theta \\ y' &= -(x - x_0)\sin\theta + (y - y_0)\cos\theta. \end{aligned} \tag{3.3}$$

One can transform from one parameterization to another by simple equations. The connection between the two parameterizations, shown in Equations 3.1 and 3.2, is

as follows:

$$\begin{aligned} A &= a^2 \sin^2\theta + b^2 \cos^2\theta \\ B &= 2(b^2 - a^2)\sin\theta\cos\theta \\ C &= a^2 \cos^2\theta + b^2 \sin^2\theta \\ D &= -2Ax_0 - By_0 \\ E &= -Bx_0 - 2Cy_0 \\ F &= Ax_0^2 + Bx_0y_0 + Cy_0^2 - a^2b^2. \end{aligned}$$

3.3.2 Fitting an ellipse to data points by minimizing the sum of squared distances

Fitting ellipses to 2D coordinate points is desired in various fields of science and engineering. In this thesis, we have fitted ellipses to sets of edge pixel coordinates. Next, we will consider the fitting of a single ellipse to the given image coordinates. Later, in Section 3.3.3, we will also introduce the problem of fitting several close-by ellipses to the image coordinates.

The least-squares-based algorithms aim to find parameters that minimize the sum of the squared distances between the given data points and the ellipse

$$\sum_{i=1}^{n} \mathcal{D}(\mathbf{x}_i; \mathbf{a})^2, \tag{3.4}$$

where $\{\mathbf{x}_i = (x_i, y_i)\}_{i=1}^n$ is the set of n data points, $\mathbf{a} = [A, B, C, D, E, F]^T$ are the ellipse parameters, and $\mathcal{D}$ is the distance metric. The distance measure can be defined in many ways. Here, we present two distance measures: geometric and algebraic distances.

Geometric distance is defined as the shortest distance between the data point $\mathbf{x}_i$ and point $\mathbf{p}$ on the curve C

$$\mathcal{D}_G(\mathbf{x}_i, C) = \min_{\mathbf{p}\in C} \|\mathbf{p} - \mathbf{x}_i\|. \tag{3.5}$$

Geometric distance is computationally expensive. The reasons are that for each data point we have to find the closest point from the curve, and the ellipse fitting is a non-linear problem. An ellipse fitting algorithm that relies on geometric distance is for example Ahn's method [62].

A more often used distance metric in ellipse fitting is based on algebraic distance, which is relatively fast to compute. The algebraic distance for the point $\mathbf{x}_i$ and the curve C defined by the conic $P(\mathbf{x}; \mathbf{a}) = 0$ is

$$\mathcal{D}_A(\mathbf{x}, C) = P(\mathbf{x}_i; \mathbf{a}), \tag{3.6}$$

i.e. the value of P at point $\mathbf{x}_i$. Numerous methods have been developed to minimize the sum of algebraic distance with ellipse-specific constraint $B^2 - 4AC < 0$. The

minimization is difficult, since the ellipse-specific constraint makes the ellipse fitting a nonlinear optimization problem. The solutions mostly rely on generic conic fitting and iterative methods, where at each iteration non-ellipses are rejected, e.g. [63, 64, 65]. In [66], the coefficients $\{A, B, C\}$ are transformed into $\{P^2, 2PQ, Q^2 + R^2\}$ to guarantee the resulting conic being an ellipse, as the ellipse-specific constraint is $B^2 - 4AC = 4P^2Q^2 - 4P^2(Q^2 + R^2) = -4P^2R^2 < 0$.

Fitzgibbon et al. [61] proposed in 1999 an ellipse-specific direct least square fitting of ellipse. The algorithm is as follows. The ellipse-specific constraint $B^2-4AC < 0$ is replaced by the equality constraint $4AC - B^2 = 1$, by using a proper scaling. The ellipse parameters can be scaled since $\alpha \cdot \mathbf{a}$ represents the same ellipse as $\mathbf{a}$. In addition, the equality constraint does not restrict the set of possible ellipses, as there are six parameters in $\mathbf{a}$ and an ellipse requires five, which means there is one free parameter that can be adjusted to fulfill the equality requirement. The equality constraint, $4AC - B^2 = 1$, can be expressed in the matrix form

$$\mathbf{a}^T\mathbf{C}\mathbf{a} = 1, \tag{3.7}$$

where constraint matrix $\mathbf{C}$ is

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{3.8}$$

The resulting ellipse-specific fitting problem is

$$\min_{\mathbf{a}} \|\mathbf{D}\mathbf{a}\|^2 \quad \text{subject to} \quad \mathbf{a}^T\mathbf{C}\mathbf{a} = 1, \tag{3.9}$$

where the design matrix $\mathbf{D}$ is

$$\mathbf{D} = \begin{bmatrix} x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i^2 & x_iy_i & y_i^2 & x_i & y_i & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & x_ny_n & y_n^2 & x_n & y_n & 1 \end{bmatrix}. \tag{3.10}$$

Applying the Lagrange multipliers, the optimal solution to $\mathbf{a}$ has following conditions

$$\begin{aligned} 2\mathbf{D}^T\mathbf{D}\mathbf{a} - 2\lambda\mathbf{C}\mathbf{a} &= 0 \\ \mathbf{a}^T\mathbf{C}\mathbf{a} &= 1 \end{aligned} \tag{3.11}$$

which can be written as a system

$$\begin{aligned} \mathbf{Sa} &= \lambda \mathbf{Ca} \\ \mathbf{a}^T \mathbf{Ca} &= 1, \end{aligned} \tag{3.12}$$

where the scatter matrix is $\mathbf{S} = \mathbf{D}^T\mathbf{D}$. The system can be solved as a generalized eigenvalue problem which results into six eigenvalue-eigenvector pairs $(\lambda_i, \mathbf{u}_i)$. There is exactly one positive eigenvalue λ_i which gives the solution $\hat{\mathbf{a}} = \mu_i \mathbf{u}_i$, where μ_i is given by $\mu_i^2 \mathbf{u}_i^T \mathbf{C} \mathbf{u}_i = 1$, i.e. $\mu_i = \sqrt{\frac{1}{\mathbf{u}_i^T \mathbf{C} \mathbf{u}_i}}$.

In this thesis, we have applied Fitzgibbon's approach to fit ellipses into specific pixel coordinates. The approach is part of the SNEF algorithm which was originally presented in Publication I. For more discussion on the SNEF algorithm, see Section 3.4.

3.3.3 Detecting multiple touching ellipses from images

We discussed above the fitting of a single ellipse to 2D data points. Unfortunately, fitting an ellipse to pixel coordinates is usually not enough in image applications. The reason is that there are often multiple ellipses that are overlapping or occluding each other and affecting the fitting results. Hence, we will next discuss approaches for detecting multiple ellipses from binary images. A typical approach to detect ellipses from images is to produce an edge image and try to fit ellipses to the edge pixels. Also, there are approaches that works with the contour of the clump resulting from the segmentation by an ordinary segmentation algorithm. One good example of such an approach is presented in [12]. In both cases, the difficulty of the ellipse detection is related to finding the respective ellipse arcs for the fitting. There might be breaks in the edge pixels sets and the ellipse detection algorithm should be able to recover from those breaks. On the other hand, arcs of different ellipses might be connected in the edge pixel set and they need to be separated. The other difficulties for multiple ellipse detection are caused by noise and improper ellipse shapes, which are especially encountered in practical applications.

One group of ellipse detection algorithms are based on Hough transform (HT) [5, 6, 7]. HT is a parametric method for geometrical shape detection from images. The detected shapes can be for example lines, circles, or ellipses. The detection of each specific geometric shape is done separately and the detection is often performed over binary edge images. HT is performed over a binary edge image such that each edge pixel is transformed into the parameters space. All the possible parameter combinations of the searched shape that could have gone through the specific edge pixel in the image space, are shown by a curve or surface in the parameter space representation of the edge pixel. Therefore, the peaks on the parameter space correspond to searched shapes that have gained support from several edge pixels. In practice, the parameter space is subdivided into regions called accumulator

cells, and each edge pixel gives one score to the cells on which its transformed curve is lying. The accumulator cells having highest number of scores represents the searched shapes in the image space. The advantage of HT-based approaches is that the approach can cope with small gaps often present in the edge image presentation of objects. A drawback of HT is that being an edge based approach, it is sensitive to texture and image noise. In addition, HT has been widely criticized in numerous papers, e.g. [8, 9, 10], to have high computational cost and its applicability to detect shapes from real images. Namely, HT needs a very precise parametric description of the shape, which in real images causes the local deformations of the shapes to generate a large number of local maxima in the parameter space [9]. Different variants of HT have been proposed, and approaches especially for ellipse detection from images include e.g. [10, 67, 68].

In recent years, more efficient approaches to multiple ellipse detection from binary edge images have been proposed, e.g. [11, 58, 59]. Compared to HT, they restrict the search space by grouping edge pixels into arcs, and then grouping the arcs based on their likelihood on belonging to the same ellipse. One of the state-of-the-art edge-based ellipse detection algorithms was introduced in 2012 by Prasad [11]. It performed the best in the experiments presented in [58]. A method designed especially for practical applications is proposed by Bai et al. [12]. The approach is designed for detecting ellipse-resembling cell nuclei from the contours of watershed segmentation results. The approach is based on contour smoothing and concave point detection, which gives smoothed ellipse arcs to which ellipses are fitted. The fitted ellipses are combined based on predefined properties for the ellipses and their fit to the ellipse arcs. Therefore, the approach needs a careful selection for the set of control parameters, as noted in the experiments presented in [11].

3.4 SNEF algorithm for segmentation of cell nuclei by ellipse fitting

An algorithm for segmentation of nuclei by ellipse fitting (SNEF) for gray level histological images is proposed in Publication I. The algorithm is designed to be fast and efficient. Therefore, the proposed algorithm consists of fairly fast and simple image processing algorithms such as thresholding (introduced in Section 2.1) and morphological operations. The proposed algorithm generates several candidate ellipses, which are later ranked and selected for the final representation by the proposed goodness-of-fit criterion. The proposed algorithm consists of three main steps. The first step combines intensity and gradient information to form an edge image that is used in the following steps. The second step finds sets of candidate ellipses for a number of seeds. The third step selects ellipses from sets of candidate ellipses for final representation. Each step of the algorithm is reviewed in more detail in the following section.

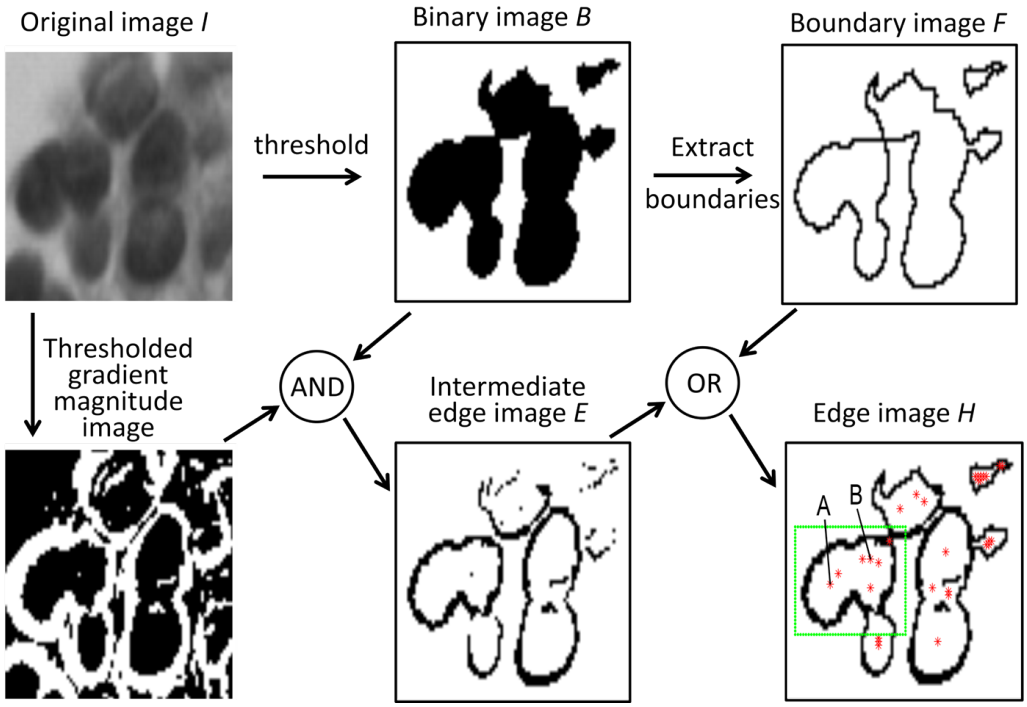


Figure 3.3: Process flow chart of the first step of the proposed SNEF algorithm.

3.4.1 Combining intensity and gradient information to form an edge image for the subsequent object detection

The first step of the proposed SNEF algorithm combines intensity and gradient information by simple and fast image processing techniques to form an edge image, H, and the corresponding edge pixel set, $\mathscr{H}_1$, which are used in the following steps of the algorithm. The processing of the first step of the proposed algorithm is visualized in Figure 3.3.

First, the original image I is thresholded to a binary image B by dual thresholding [32]. As discussed in Section 2.1, thresholding is a fast and computationally efficient image processing method to produce preliminary image segmentation results. The reason to use dual thresholding here is that, in general, histological images contain three tissue components: cell nuclei, cytoplasm, and background. One could start the object detection already from the binary image B, or its boundary image F. However, more precise estimates for the object and clump boundaries can be obtained by gradients. In addition, the gradients within the clump of objects may help in the clump splitting. We estimate the gradients by the Sobel operator [1], which is convolved with the original image I and gives as the result the gradient magnitude image G. The introduction to the gradient magnitude estimation is given in Section 2.2. To maintain only strong edges, the gradient magnitude image G is thresholded by dual thresholding. Since the

thresholded gradient magnitude image contains gradients outside the areas of the binary image B, and we are only interested in the gradients on the border and inside the possible clump, the thresholded gradient magnitude image is combined with image B by the AND operator, resulting in the intermediate edge image E. The problem with the intermediate edge image E is that it does not guarantee closed borders. Especially, in case of nuclei slowly fading to the background, there is no gradient on the side of the fading. To guarantee closed borders, the image E is combined with the boundary image F by using the OR operation. The resulting image is cleared by removing small isolated regions, after which the image is denoted as the edge image H, and its corresponding pixel set is $\mathscr{H}_1$.

3.4.2 The search for the sets of candidate ellipses

The proposed SNEF algorithm generates several candidate ellipses from which the ellipses of the final representation for the clump are selected in the later step. Therefore, the algorithm tolerates having more seeds and ellipses than there will be in the final representation. The sets of candidate ellipses are found by using the edge image H, generated in the previous step. The main phases for finding the set of candidate ellipses consist of finding the seeds, rotating a ray centered at the seed and picking pixels from the edge set $\mathscr{H}_1$, grouping pixels, and fitting ellipses to the grouped pixel sets. The problem of splitting clumps has been discussed in Section 3.2, and the problems of splitting clumps by ellipses in particular are presented in Section 3.3.3.

The search for the sets of candidate ellipses starts by finding seeds. The set of seeds $\mathscr{S}$ are found by ultimate erosion, which is a popular approach for obtaining seeds often required in segmentation algorithms. Ultimate erosion is a morphological image processing operation in which an object in a binary image is eroded until the object disappears. During the erosion process, the eroded object may split into smaller non-connected objects, out of which some might disappear earlier than the others. The output of the ultimate erosion consists of the pixels that are the last remaining pixels of each of the non-connected parts of the object just before it disappears. Therefore, the pixel sets resulting from ultimate erosion are in practice the local maximas of the distance transformed object image. The distance transform is in general calculated such that for each object pixel, the distance to the closest background pixel is calculated by using some distance metric. We applied ultimate erosion to the edge image H, and hence our approach differs slightly from traditional approaches in which the object is usually presented as a set of connected pixels, such as the binary image B, which is shown in Figure 3.3. In our case, the background pixels consisted of the edge pixel set $\mathscr{H}_1$, and the object pixels were the ones inside the edge pixel set. As the distance metric, we used the Euclidean distance.

The set of candidate ellipses associated to a seed $S_i = (x_{0i}, y_{0i}) \in \mathscr{S}$ is obtained as follows. First, a ray is rotated centered at a seed. At each angle $\alpha \in \{1^0, \ldots, 360^0\}$,

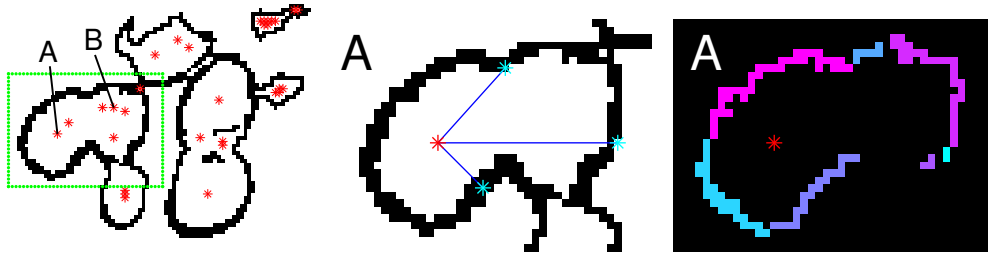


Figure 3.4: Finding seeds and connected components. (a) The edge image H, and the seeds $\mathscr{S}$ (red stars) resulting from ultimate erosion applied to the edge image H. (b) A ray centered at the seed A (red star) rotates, picking at each angle one pixel from the edge pixel set $\mathscr{H}_1$ (only three ray positions are shown). (c) Edge pixels picked by the ray are grouped into seven connected components (each connected component has its own color). The figures are originally presented in Publication I, first published in the Proceedings of the 18th European Signal Processing Conference (EUSIPCO-2010) in 2010, published by EURASIP.

the closest pixel on the line $x_i = x_{0i} + r\cos\alpha$, $y_i = y_{0i} + r\sin\alpha$, $r > 0$ that belongs to the edge pixel set $\mathscr{H}_1$, is appended to the pixel set $C_\alpha(x_{0i}, y_{0i})$. The purpose of the rotating ray is to find from the edge pixels set $\mathscr{H}_1$ the pixels that could be considered as the border pixels of the cell nuclei associated to the seed, and the locations of the disconnections in the pixel set $C_\alpha(x_{0i}, y_{0i})$ are the locations of possible concave points on the edge or other discontinuities. Therefore, the pixels at the set $C_\alpha(x_{0i}, y_{0i})$ are grouped into connected components denoted as $\mathscr{C}_1, \ldots, \mathscr{C}_{n_c}$ so that there is no need to fit ellipses to all the possible pixel combinations; instead, the ellipses can be fitted to the pixel sets made from connected components. In addition, we assume that the closer the pixel in the set $C_\alpha(x_{0i}, y_{0i})$ is to the seed, the more likely it belongs to the nuclei border associated to the seed. Hence, the connected components are arranged into increasing order starting from the smallest distance from the connected component to the seed $S_i = (x_{0i}, y_{0i})$, and incrementally appended to the set $\mathscr{D}$. After each addition to the set $\mathscr{D}$, we fit an ellipse to the pixel coordinates in the set $\mathscr{D}$. The used ellipse fitting algorithm is Fitzgibbon's direct least square fitting of ellipses method introduced in [61] and discussed in Section 3.3.2. The method is selected because it directly fits an ellipse to pixel coordinates; therefore, there is no need for iteration to find the optimal ellipse parameters. The resulting ellipse parameters from the Fitzgibbon method are used to generate the ellipse pixel set $\mathscr{E}(S_i, \ell)$, where ℓ corresponds to the stage of connected component increment, i.e. the number of appended connected components at the set $\mathscr{D}$, and the set $\mathscr{E}(S_i, \ell)$ contains the pixels that are on the ellipse curve.

As a result, we have altogether a number of n_c candidate ellipses associated to the seed S_i. In addition, some of the seeds can be close-by and represent the

same nuclei. Thus, ellipses for the final representation need to be selected. We proposed in Publication I a goodness-of-fit criterion for ordering and selecting the ellipses. The proposed criterion is reviewed in the next subsection.

3.4.3 Selecting the final ellipses from sets of candidate ellipses by a relatively simple goodness-of-fit criterion

The selection of the ellipses for the final representation is done on two levels by the proposed goodness-of-fit criterion. In the first level, the candidate ellipses within each seed are competing, and for each seed, the ellipse having the highest value of criterion is chosen. In the second level, the ellipses selected at the first level are ordered, with those having a higher value of criterion and not having a more than 60% overlapping area with an already selected ellipse being kept.

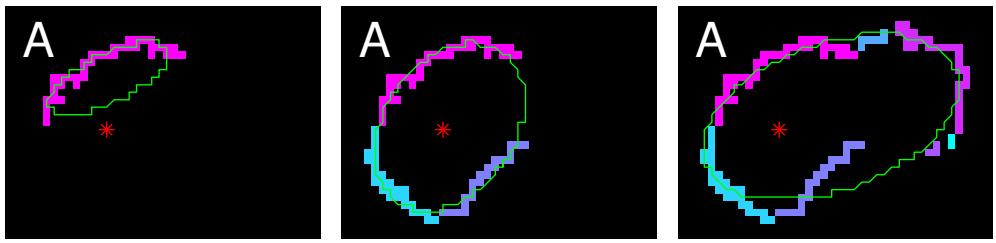
The proposed goodness-of-fit criterion for comparing and selecting ellipses is given by

$$V(S_i,\ell) = V_1(S_i,\ell) + V_2(S_i,\ell) = \frac{|\mathscr{E}(S_i,\ell) \cap \mathscr{H}_1'|}{|\mathscr{E}(S_i,\ell)|} + \frac{|\mathscr{D}(S_i,\ell) \cap \mathscr{E}'(S_i,\ell)|}{|\mathscr{D}(S_i,\ell)|}, \quad (3.13)$$

where V_1 and V_2 denotes the first and second terms of the criterion V; $\mathscr{H}_1$ denotes the edge image; $\mathscr{E}(S_i,\ell)$ denotes the set of ellipse pixels generated by using ellipse parameters $\Theta(S_i,\ell)$ and rounding the obtained coordinates to the resolution of the image grid; $\mathscr{D}(S_i,\ell)$ denotes connected components appended into set $\mathscr{D}$ at stage ℓ; and the prime denotes dilation of the image with a structuring element, which allows not only for the exact matching of pixels, but also the matches in the four neighborhood vicinity.

The criterion balances between two terms; first, fitting of the ellipse pixels to the edge pixels, $\mathscr{H}_1$, and second, the pixels on the current set of connected components, $\mathscr{D}(S_i,\ell)$, fitting to the ellipse pixels. The first term, $V_1(S_i,\ell)$, is high once the ellipse pixels are most of the time on the edge pixels, $\mathscr{H}_1$, and only rarely over the background. In case of overlapping nuclei, an ellipse most probably has difficulties getting the highest first term values, since there may not be any edge pixels within the nuclei. The second term, $V_2(S_i,\ell)$, gets most likely high values when the connected components have a low number of pixels, i.e. the connected component forms a short concave curve and it is relatively easy to fit an ellipse to the curve.

The performance of the proposed criterion is visualized for the seed A in Figure 3.5. There are shown three different stages, ℓ, of connected components set $\mathscr{D}$ to which ellipses are fitted. In the first stage, there is only one connected component in the set $\mathscr{D}(A,1)$. The ellipse fitted to the pixel coordinates in the pixel set $\mathscr{D}(A,1)$ agrees well with the pixel set so that the second term of the criterion is high, being $V_2(A,1) = 0.97$. However, the first term of the criterion is low, being $V_1(A,1) = 0.68$. The reason is that the ellipse is only about half of the time on



(a)	(b)	(c)
$V(A,1) = 1.65,$	$V(A,3) = 1.85,$	$V(A,7) = 1.74,$
$V_1(A,1) = 0.68,$	$V_1(A,3) = 0.86,$	$V_1(A,7) = 0.96,$
$V_2(A,1) = 0.97.$	$V_2(A,3) = 0.99.$	$V_2(A,7) = 0.78.$

Figure 3.5: Visualization for the performance of the goodness-of-fit criterion presented in Equation 3.13 for the seed A (red star). The values of the criterion, V, and its respective terms V_1 and V_2 are presented under each sub-figure. (a) The closest connected component (pink pixels) to the seed (red star) and the ellipse (in green) fitted to the pixels of the connected component. (b) The three closest connected components (each connected component has its own color) closest to the seed and the fitted ellipse (in green). (c) All the connected components found by the rotating ray centered at the seed A (red star), and the fitted ellipse (in green). The figure is originally presented in Publication I, first published in the Proceedings of the 18th European Signal Processing Conference (EUSIPCO-2010) in 2010, published by EURASIP.

the edge pixels $\mathscr{H}_1$. In the second stage, the number of connected components in the set $\mathscr{D}(A,3)$ is three, and the value of the first term has improved from the first stage, now being $V_1(A,3) = 0.86$. And in the third stage, all the seven connected components are added to the set $\mathcal{D}$, and the value of the first term is the best, being $V_1(A,7) = 0.96$. However, the second term is the worst, being $V_2(A,7) = 0.78$. This is because the ellipse is not fitting as well as the others to its respective set of connected components $\mathscr{D}(A,7)$. As a result, the highest value of the criterion is obtained for the seed A by the second stage ellipse which was resulting from the ellipse fitting to the three connected components.

Figure 3.6 visualizes the selection of the ellipses for the final segmentation. For each seed, there are multiple fitted ellipses, see Figures 3.6 (c) and (e). The best fitting ellipses is selected based on the criterion presented in Equation 3.13, shown in red in Figures 3.6 (c) and (e). Therefore, there is one ellipse proposal from each seed for the final segmentation. The proposal ellipses are ordered by criterion, since there might be some highly overlapping ellipses resulting from close-by seeds and it is tolerated to have more seeds than ellipses in the final representation. Then, we have selected from the ordered list only the ones that are not overlapping more than 60% with a better fitting ellipse.

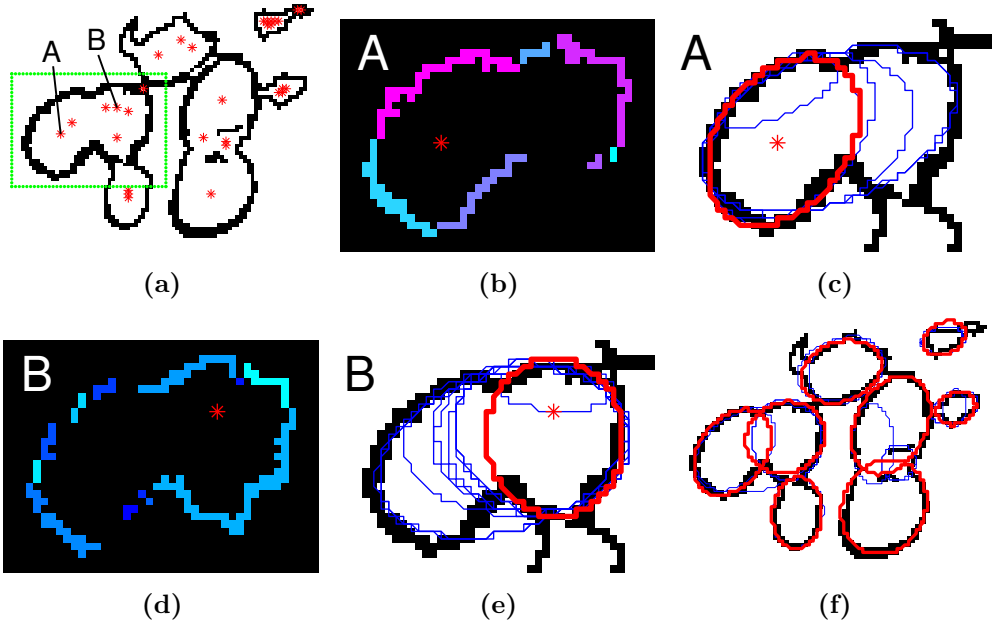


Figure 3.6: Selecting the ellipses for the final segmentation. (a) The edge image H, and the seeds $\mathscr{S}$ (red stars) resulting from ultimate erosion applied to the edge image H. The locations of two seeds, A and B, are highlighted. (b) The pixels of the connected components resulting from the rotating ray centered at the seed A (c) Ellipses (in blue) resulting from ellipse fitting to the sets of connected components. The best ellipse based on the criterion presented in Equation 3.13 is shown in red (d) The pixels of the connected components resulting from the rotating ray centered at the seed B. (e) Ellipses (in blue) resulting from ellipse fitting to the sets of connected components. The best ellipse based on the criterion presented in Equation 3.13 is shown in red. (f) The final ellipse representation for the clump of cell nuclei (in red). The blue ellipses are the best ellipses of some seeds, but discarded from the final representation due to overlapping some better fitting ellipses. The figures are originally presented in Publication I, first published in the Proceedings of the 18th European Signal Processing Conference (EUSIPCO-2010) in 2010, published by EURASIP.

3.4.4 Results and discussions

The proposed SNEF algorithm is tested on a histological image consisting of several clumps of cell nuclei. The results presented in Publication I show that proposed algorithm is able to segment and split clumps in relatively clear cases. The advantages of the algorithm are that is designed to be fast and efficient to propose numerous ellipses that are at least partly on edge, or a clump border. The main restriction of applying the algorithm to a wide range of histological images, and images in general, concerns the selection of two thresholding methods, the intensity image thresholding and gradient image thresholding, when the intensity image and gradient image information is combined to an edge image H. As discussed in Section 2.1, there are several thresholding methods available, and

since they emphasize different image features, the results of different segmentation methods differ. On the other hand, the thresholding is applied in the proposed algorithm due to its speed in giving preliminary segmentation results. The problem of finding the correct thresholding method can be alleviated by a more refined clump splitting or interpretation selection criterion which has information theoretic grounds and is based on the minimum description length (MDL) principle [20]. The MDL-based approach will be discussed in the next chapter.

4 Information theoretical approach to segmentation

The previous chapters introduced methods and challenges of segmentation and separation of overlapping objects in images. In this chapter, an information theoretical approach to image segmentation and interpretation of signals is taken. The special interest is on the minimum description length (MDL) principle based approaches. Publications II, III, and IV will be reviewed and discussed within the chapter.

The structure of this chapter is as follows. First, we give an introduction to model selection. Then, the connection of fundamental concepts of information theory: coding and entropy, are presented. After that, the minimum description length (MDL) [20] principle is reviewed, which includes a more detailed introduction to two different implementations of MDL: two-part coding based MDL [20], and one of the most recent implementation, sequentially normalized maximum likelihood (SNML) models [24, 25]. The two-part coding based MDL is the oldest version of MDL, and widely applied in image segmentation. The SNML is applied in Publication IV to detect changes in time series data. The rest of the chapter concentrates on MDL based image segmentation, and clump splitting. First, an introduction to MDL-based image segmentation is given. Then, we will review a two-part coding based image segmentation approach. Finally, an introduction to the proposed MDL-based ranking for competing interpretations of a clump is given. In addition, the experimental results presented in Publications II and III are introduced.

4.1 An introduction to model selection

Models are mathematical formulas used to describe a studied phenomenon. The model construction usually starts with having some observations which are often measured together with some explanatory variables. When designing models, one should use all the available knowledge. In physical models, the connection between the explanatory variables and observations is deterministic. However, in many cases a deterministic model does not cover all the aspects of the phenomenon.

Statistical models can be used on those cases on which a deterministic model does not cover. In statistical models, the regularities between the observations and explanatory variables can be modeled by probability distributions.

Model selection aims to select between competing statistical models the one that best describes the studied phenomenon. One of the main challenges encountered in model selection is over-fitting: the model is extremely complex and its ability to generalize to the unseen data from the same source is reduced. Occam's razor is often referred to as the first model selection approach. It favors simple theories over unnecessarily complex ones. One commonly preferred approach for model selection and validation is to divide the data into distinct training, test, and validation sets. However, the number of data samples is often restricted, and more effective model selection approaches are needed. Therefore, several parametric and non-parametric model selection approaches have been developed. In non-parametric approaches, such as cross-validation (CV) [16] and bootstrapping [17], the studied data set is sampled multiple times into training and test sets. For instance, leave-one-out (LOO) is one version of CV. In LOO, each data sample is in turn left out from the training set and used as a test set. Akaike's information criterion (AIC) [18], Bayesian information criterion (BIC) [19], and the minimum description length (MDL) principle [20, 21] are examples of the parametric model selection approaches. They utilize the whole data set of samples for measuring the goodness of the model by measures such as likelihoods of the data, given the model, and penalizes the number of model parameters, such that too complex models are not favored. The penalization terms and the philosophies behind the parametric model selection approaches differ.

Next, we will describe important preliminaries for information theory based model selection. Namely, we will describe important connections between codelength, probabilities and Shannon entropy. After that, we will introduce the MDL principle in more detail.

4.2 Coding, probability and entropy

The roots of the MDL principle are on information theory. Therefore, the fundamental concepts of information theory: coding, probability, and entropy; and their important connections are next introduced. A thorough introduction to information theory can be found in the book written by T. Cover and J. Thomas [69].

A code, C, for a finite or countable set, e.g. an alphabet, $\mathcal{X}$, is defined as a mapping from $\mathcal{X}$ to the set of codewords, which are usually binary strings, i.e. sequences of 0s and 1s. The encoding of x with a given code C is denoted as $C(x)$, where x is symbol or discrete real-valued number from the set $\mathcal{X}$. Typically, a data sequence $x^n = x_1, \ldots, x_n$ is before storing or transmission encoded into a sequence of codewords, which forms a finite binary string. The sequence of

codewords is formed by concatenating the codewords of the corresponding symbols of the data sequence, i.e. $C(x_1), \ldots, C(x_n)$. However, the concatenated sequence of codewords is not necessarily decodable without introducing an extra symbol for separating the codewords. The decodability of the sequence is guaranteed by prefix codes, in which no codeword is allowed to be a prefix of another codeword. Hence, prefix codes allow one-to-one mapping from a data sequence into sequence of codewords, and vice versa. In addition, no extra symbols are needed to introduce to the alphabet in order to separate concatenated codewords.

The codelengths of the prefix codes obey the Kraft's inequality. If the codes are prefix codes, the inequality is satisfied by

$$\sum_{x \in \mathcal{X}} 2^{-l(x)} \leq 1, \tag{4.1}$$

where $l(x)$ is the length of the codeword $C(x)$ and the summation is over all the elements in $\mathcal{X}$. In addition, the equality holds if and only if the code is complete, in which case all the leaves of the prefix tree are codewords.

In practice, when constructing the encoding scheme and assigning codelengths, one should aim to minimize the expected mean length of the message codewords. The average length of the codewords is given as

$$L = \sum_{x \in \mathcal{X}} p(x) l(x), \tag{4.2}$$

where $p(x)$ is the probability of the occurrence of the symbol $x \in \mathcal{X}$. The optimal codelengths, minimizing the average codelength, L, are

$$l^*(x) = -\log_2 p(x). \tag{4.3}$$

Hence, in the optimally constructed coding scheme, short codelengths correspond to high probabilities, and vice versa. The resulting average length of the optimal codewords is

$$L^* = -\sum_{x \in \mathcal{X}} p(x) \log_2 p(x), \tag{4.4}$$

which is Shannon entropy. Unfortunately, the optimal codelengths $l^*(x)$ might not be integers. The optimal integer valued codelengths and a corresponding prefix code can be found by the Huffman algorithm [70], which gives the average length of the codewords within 1 bit of the entropy. In addition, we rarely have the actual probabilities of the occurrences of the symbols. Often, the probabilities of the occurrences need to be estimated, and the closer the estimated probabilities are to the true ones, the better performance one can achieve.

4.3 The minimum description length principle

The minimum description length (MDL) principle [20] is an information theory based approach to model selection, estimation, and statistical inference. The

main idea of MDL is to find regularity in the given data. The regularity can be identified by compression: the more regular the data sequence is, the more it can be compressed. Therefore, learning the data is equated with compressing the data. The MDL principle differs essentially from the other model selection approaches in a sense that the MDL does not assume that the data would have been generated according to some distributions. However, if such a data generating mechanism exists, it provides the minimum description for the data as stated with Shannon entropy. MDL has been mostly thought as a model selection approach. However, it has also been successfully applied to numerous applications in various fields including denoising [71, 72], clustering [73], and DNA sequence modeling [74].

The MDL principle has some connections to other parametric model selection approaches: Akaike's information criterion (AIC) [18] and Bayesian information criterion (BIC) [19]. AIC is the first model selection method that relies on information theory. The formula of BIC, on the other hand, coincides under certain conditions with an MDL-based criterion, which has led to persistent beliefs that BIC and MDL would be the same. Nevertheless, the MDL principle essentially differs from BIC and AIC in the method and the underlying philosophy levels. For instance, in BIC, the Bayesian prior represents the uncertainty of parameters before the observations, while in MDL the corresponding 'prior' stems from the needs of being able to encode the parameters.

A more detailed introduction to the MDL principle can be found in a book written by P. Grünwald [75], and the older book [76]. In addition, J. Rissanen has written numerous books that consider MDL; the most recent ones being [77, 78]. Next, we will present an introduction to different embodiments of the MDL principle. Then, we will describe two MDL approaches used in this thesis: two-part coding and sequentially normalized maximum likelihood models.

4.3.1 From ideal MDL to practical MDL

Ideal MDL has its roots in theory of Kolmogorov complexity [79], which was developed by Kolmogorov [22], Chaitin [80], and Solomonoff [81, 82]. The Kolmogorov complexity of a sequence is defined as the length of the shortest program by a universal computer language that prints the sequence and then halts. Therefore, Kolmogorov complexity measures the regularity of the sequence: more regular and less random sequences obtain lower values of Kolmogorov complexity, since fewer bits are needed to encode the sequence. The invariance theorem states that the difference between description lengths of two different universal languages for a data sequence is negligible compared to the length of the data sequence as long as the data sequence is large (asymptotically). Unfortunately, Kolmogorov complexity is uncomputable [79]. First, there is no computer program that for every sequence of data returns the shortest program that prints the data and then halts. Second, we are many times interested in data sets having a small

number of samples. In such data sets, the invariance theorem does not hold, and the complexity is dependent on the syntax of the computer language.

Practical MDL aims to implement the ideas of Kolmogorov complexity by using less expressive description methods than used in universal computer languages [75]. The description methods are compromising between generality and restriction, so that we could compress many of the regular sequences and at the same time being always able to obtain the length of the shortest description of any data sequence. The drawback of practical MDL is that there will be regular sequences that cannot be compressed. There are four main implementations of the practical MDL. The earliest implementations are based on two-part codes [20]. The three more recent implementations are based on minimax optimal universal models, which are probability distributions corresponding to universal codes [83], and includes Bayesian mixture code [84], normalized maximum likelihood codes (NML) [21, 23], and sequentially normalized maximum likelihood codes (SNML) [24, 25]. Normalized maximum likelihood (NML) [21, 23], originally proposed by Shtarkov [23] in 1987 and connected to the theory of MDL by Rissanen [21] in 1996, is probably the most used and studied approach. The main problems of the NML are that it requires hyperparameters and the normalization coefficient is difficult to calculate in practice. The sequentially normalized maximum likelihood (SNML) universal model [24, 25] is a more recent implementation and it is proposed to circumvent the computational problems of the NML. In this thesis, we concentrate on two-part codes and SNML models. The two main reasons are that most of the image segmentation papers written so far are based on two-part codes, and that SNML is especially applicable for time series data modeling, which is the other main objective of this thesis.

4.3.2 Two-part coding based MDL

The earliest and simplest implementation of MDL is the so-called two-part code [20] introduced in 1978 by J. Rissanen. Although the theory and methods of MDL have evolved, two-part code can be the only applicable approach in certain applications. In the field of image segmentation, most of the papers are based on two-part coding. In addition, two-part coding is a good approach to introduce the main ideas of the MDL principle.

The idea of the two-part MDL is to choose the model M which minimizes the total codelength for both the data D and the model M. The total codelength is given by

$$L(D, M) = L(D|M) + L(M), \tag{4.5}$$

where L denotes codelength, $L(D|M)$ is the codelength for encoding the data D with the help of the model M, and $L(M)$ is the codelength for encoding the model M. The codelength $L(D|M)$ describes how close the observed data is to the assumed model; the smaller the term, the better the model fits the data, and vice versa. This results from the fact that we have to encode only the errors the

model makes on the data instead of the full data. Usually, the better the model fits the data, the more complex models are needed, and the more bits are needed to encode the model, i.e. $L(M)$ grows. Hence, the minimized total codelength $L(D, M)$ is a compromise between goodness-of-fit and model complexity.

4.3.3 Sequentially normalized maximum likelihood models

Sequentially normalized maximum likelihood (SNML) models [24, 25, 85] are one of the most recent approaches to implement MDL. Most of the papers written thus far on SNML are for linear regression models with the assumption of Gaussian distributed residuals. Therefore, the approach has also been called the sequentially normalized least squares (SNLS) model [24, 25]. The advantages of the SNML models over normalized maximum likelihood (NML) [21, 23] universal models include that it is computable for autoregressive (AR) and autoregressive–moving-average (ARMA) models, and there is no need for hyperparameters in the normalization of density functions.

Let a data vector be $y^n = [y_1, \dots y_n]'$, and modeled by linear regressions

$$y_t = b_t' \bar{x}_t + \hat{e}_t = \sum_{i=1}^{k} b_{t,i} x_{t,i} + \hat{e}_t, \tag{4.6}$$

where $\bar{x}_t = [x_{t,1}, \dots, x_{t,k}]'$ are the columns of the regressor matrix X_t, $b_t = [b_{t,1}, \dots, b_{t,k}]'$ are the model parameters, and $\{\hat{e}\}_{t=1}^n$ are assumed to be independent and identically distributed (i.i.d.) sequence from Gaussian distribution of zero mean and variance σ^2. When modeling the sequence of observations y^n with kth order AR model, the columns of the X_t are $\bar{x}_t = [y_{t-1}, \dots, y_{t-k}]'$. In SNML, the parameter estimates b_t are estimated using data available up to t. Therefore, it differs from the traditional least squares approach in which all the n observations are used, and the estimated parameters b_t are the same for all the $t = 1, \dots, n$. In addition, the approach differs from the so-called 'plug-in' predictor and the related predictive least squares (PLS) criterion [86, 87], in which the parameter estimates b_t are estimated from the data available up to $t - 1$.

The idea of SNML is that the minimized sum of squared residuals is calculated recursively [24, 25]:

$$\hat{s}_t = \hat{s}_m + \sum_{j=m+1}^{t} (y_j - \bar{x}_j' b_j)^2 = \hat{s}_m + \sum_{j=m+1}^{t} \hat{e}_j^2 = \hat{s}_{t-1} + \hat{e}_t^2, \tag{4.7}$$

where $t > m$ and m is the smallest fixed number for which the maximum likelihood estimate can be computed. The sum of squared residuals of SNML is smaller than the sum obtained by the traditional least squares approach or the predictive approach [24, 25]. Unfortunately, the minimized sum of squared deviations $\hat{s}_t/t$

cannot be directly used in the function

$$q(y_t|y^{t-1}, X_t; \sigma^2, b_t) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(y_t - \hat{y}_t)^2}{2\sigma^2}\right), \tag{4.8}$$

as in the case of fixed variance. The reason is that the normalization of the function would require an integral, which gives infinity as a result [24, 25]. Therefore, SNML considers the following maximization problem

$$\max_{\sigma^2} \prod_{t=m+1}^{n} f(y_t|y^{t-1}, X_t; \sigma^2, b_t), \tag{4.9}$$

and the conditional density and the normalized conditional density functions, $f(y_t|y^{t-1}, X_t)$ and $\hat{f}(y_t|y^{t-1}, X_t)$, defined in [24] as

$$f(y_t|y^{t-1}, X_t) = \frac{f(y^t|X_t)}{f(y^{t-1}|X_{t-1})}, \tag{4.10}$$

$$\hat{f}(y_t|y^{t-1}, X_t) = \frac{f(y_t|y^{t-1}, X_t)}{K(y^{t-1})}, \tag{4.11}$$

$$K(y^{t-1}) = \int f(y_t|y^{t-1}, X_t)\mathrm{d}y_t. \tag{4.12}$$

The SNML model is obtained by multiplying the normalized conditional density functions and an initial density function $q(y^m|X_m)$. The SNML model is given in [24, 25] as

$$\hat{f}_{\mathrm{SNML}}(y^n|X_n) = q(y^m|X_m) \prod_{t=m+1}^{n} \hat{f}(y_t|y^{t-1}, X_t). \tag{4.13}$$

The criterion for the model selection is the negative logarithm of the SNML model

$$\mathrm{SNML}(n, k) = -\ln \hat{f}_{\mathrm{SNML}}(y^n|X_n). \tag{4.14}$$

With the solution for σ^2 presented in Equation 4.9:

$$\hat{\tau}_n = \frac{\hat{s}_n - \hat{s}_m}{n - m} = \frac{1}{n - m} \sum_{t=m+1}^{n} \hat{e}_t^2, \tag{4.15}$$

the result of the maximized product is $(2\pi e\hat{\tau}_n)^{-(n-m)/2}$, and the resulting sequentially normalized least squares (SNLS) criterion for model selection is

$$\begin{aligned} \mathrm{SNLS}(n, k) &= -\ln \hat{f}_{\mathrm{SNML}}(y^n|X_n) \\ &= \frac{n - m}{2} \ln(2\pi e\hat{\tau}_n) + \sum_{t=m+1}^{n} \ln(1 + c_t) + \frac{1}{2} \ln n + O(1), \end{aligned} \tag{4.16}$$

where $c_t = \bar{x}_t' V_{t-1} \bar{x}_t$ and in there $V_t = (X_t X_t')^{-1}$ come from the recursions presented in [88]. In our experiments presented in Publication IV, we used the following expression for the model selection criterion

$$-\ln \hat{f}_{\mathrm{SNML}}(y^n|X_n) = \frac{n}{2}\ln(2\pi e \hat{s}_n/n) - \sum_{t=m+1}^{n} \ln(1-d_t) + \frac{1}{2}\ln n + O(1). \quad (4.17)$$

The reason for the differences in Equations 4.16 and 4.17 is that we used $\hat{\sigma}_n^2 = \hat{s}_n/n$ as a solution for Equation 4.9. In addition, the second terms in Equations 4.16 and 4.17 are the same, since $1 - d_t = 1/(1 + c_t)$, as shown in [86, 87].

4.4 Segmentation and interpretation of time series data by MDL

In Publication IV, the sequentially normalized maximum likelihood (SNML) is applied to time series analysis, and the monitoring of changes in the measured signal. Changes in a time series signal can be induced, for instance, by changes in machine condition or incipient machine failure. MDL provides an efficient framework for machine condition monitoring and signal change detection, since the changes in the measured signal affects the signal complexity. MDL is especially designed to measure the signal complexity: when the signal changes, the description and codelength needed to describe the signal also changes. Autoregressive (AR) models are widely used in time series modeling and signal change detection. Therefore, in Publication IV, we combine SNML with an AR model. The SNML-based model selection criterion for AR models is described in Section 4.3.3 and presented in Equation 4.17. The resulting MDL-based descriptions can be used to segment and interpret time series data for possible machine condition failures.

Next, we will describe our signal change detection and machine condition monitoring algorithm that has been originally published in Publication IV. First, the measured signal is split into smaller segments. We use a windowing technique, in which the consecutive segments are maximally overlapping such that only the samples within both ends of the segments differ. We studied the signals using multiple window sizes, which allowed us to obtain changes from different time scales. The small window sizes are detecting short-term and large windows long-term changes, respectively. For each segment, both the value of the minimized SNML-based model selection criterion and the corresponding AR model order are computed. Since we have windowed the signal with multiple window lengths, we have for each signal sample (i.e. time step) multiple corresponding segments. Therefore, the values of the model selection criteria of the segments can be concatenated into feature vectors, which can be used to classify, segment, and interpret the state of the signal by using a proper clustering or classification algorithm. In our case, we use self-organizing maps (SOM) [89]. The advantage of using SOM is that it is a vector quantization method which represents multidimensional inputs in a

two-dimensional grid with a topological preservation, i.e. the close-by map units are also nearby in the input space. Therefore, SOM provides a map from which one can visually evaluate the state of the signal, as each signal sample can be placed to the closest SOM map unit.

4.4.1 Data sets and experimental results

In Publication IV, we have used two different data sets. The first data set is a laboratory test, in which we occasionally corrupted the measured signal by an external source. The aim of the first data set is to learn basic behaviors of the SNML in signal change detection. The total length of the signal is 32 seconds long which results in 6390 samples with the sampling rate of 200Hz. The used window sizes were 50, 100, 250, and 750 samples. In the second experiment, we applied the proposed machine condition monitoring algorithm for detecting different ball bearing faults. The second data set consists of measurements of real ball bearing faults by a piezoelectric accelerometer. The measurements were performed from three different axes, which were called the vertical, axial, and horizontal axis. The failure types were inner ring, outer ring, and ball failures. Each failure signal consists of 4096 data samples, and the used window sizes were: 64, 128, 256, 512, and 1024 samples.

The results of the first experiment are shown in Figure 4.1. It can be seen that the values of the SNML-based model selection criterion provides good estimates for the signal complexity, as the places of high values of the model selection criterion are corresponding with the occasional signal corruption. The length of the windowing affects the results such that the smaller the window size, the more "noisy" the values of the model selection criterion, and the wider the window, the smoother the values of the criterion. With the widest window size, the model selection criterion is also detecting phenomena other than the actual occasional corruption. Therefore, one should use a collection of different window lengths to detect phenomena on different time scales. The AR model orders selected by the criterion do not show similar correlations with the places of external signal corruption.

For the second experiment, we have shown the SOM classification results for the vertical axis measurements in Figure 4.2. The U-matrix representation of the learned SOM, presented in Figure 4.2(a), shows the distance between the neighboring units by coloring: the higher the value (red, yellow or cyan) the more distant the units are. It can be seen that there are two distinct regions. The locations of the best matching units for the samples from the three failure types (inner, outer and ring failures) are shown in Figures 4.2(b-d). The bigger the dot within the unit, the more samples are located into that specific unit. Map units without any samples are colored on white. From Figures 4.2(b-d), it can be seen that the inner and outer ring failures are the most apart from each other, while the ball failures are in the middle and slightly overlapping with outer ring failures.

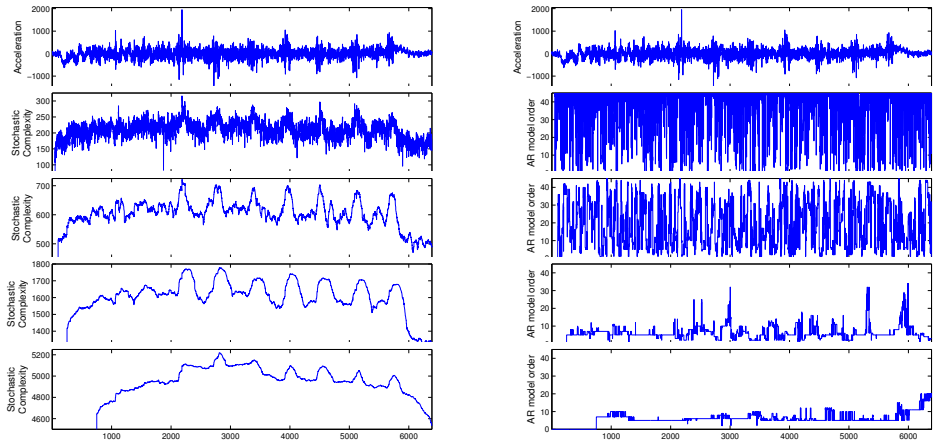
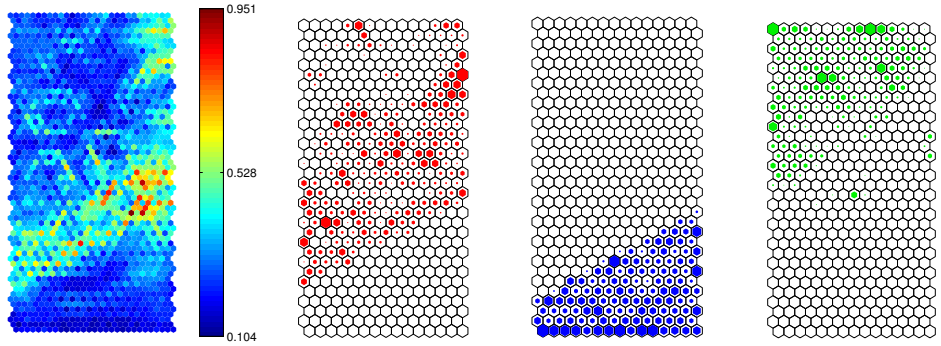


Figure 4.1: Results of the first time series experiment. Top-row: the original time series signal. Left: the values of the optimized SNML criterion. Right: the corresponding values of the estimated model orders. From the second row downwards: the used window sizes are 50, 100, 250, and 750. The figures are originally published in Publication IV. © 2008 IEEE.



(a) U-matrix **(b)** balls **(c)** inner ring **(d)** outer ring

Figure 4.2: SOM classification results for three failures (balls, inner ring, and outer ring) measured from the vertical axis of ball bearings. The coloring in the U-matrix shows the distance between the neighboring units. The best matching units for samples from each failure type are shown in (b-d). The bigger the dot within the unit the more samples are located into that unit. The figure is originally published in Publication IV. © 2008 IEEE.

We classified the samples so that each unit was labeled according to failure that had the highest number of samples on that specific unit. The classification results for all three axes, vertical (V), axial (A), and horizontal (H), are summarized in confusion matrices presented in Table 4.1. A similar behavior can be observed as in Figure 4.2. The outer ring failures can be separated from the inner ring failures. The ball failures are overlapping with inner and outer ring failures. The results

Table 4.1: Confusion matrices for the SOM classification results. The classification is done separately for the measurements from three different axes: vertical (V), axial (A) and horizontal (H). The values of the confusion matrices are expressed in percentages. The matrix is originally published in Publication IV. © 2008 IEEE.

		Predicted		
	Actual	balls	inner	outer
V axis	balls	96.0	0	4.0
	inner	0	100	0
	outer	3.6	0	96.4
A axis	balls	87.3	12.7	0
	inner	9.3	90.7	0
	outer	0	0	100
H axis	balls	87.6	8.4	4.0
	inner	3.8	96.2	0
	outer	4.1	0	95.9

of the second experiment imply that the SNML based model selection criterion provides good features for separating the different ball bearings faults.

4.5 Image segmentation based on the MDL principle

The first image segmentation paper considering the MDL principle was proposed in 1989 by Leclerc [26]. The paper was inspired by two-part coding, as most of the MDL-based approaches proposed to image segmentation [27, 28, 36]. We will describe the two-part coding based image segmentation criterion in more detail in Section 4.5.1. Now, we will shortly describe the general idea behind the two-part coding based image segmentation approaches. The idea is to minimize the total codelength of encoding the segmentation and the encoding of the pixel values of the segmentation regions such that each region is encoded separately. A popular choice for encoding the segmentation is to use chain codes. Chain codes represent the contours of the region boundaries such that each element in the chain represents the direction of the next step. The intensity variations within the segmentation regions are assumed to be either constant or piece-wise smooth and modeled by low-order polynomials, as in Leclerc's approach. The noise is assumed to be uncorrelated and Gaussian distributed.

A general problem with MDL-based image segmentation algorithms is that an exhaustive search for finding the global minimum of the MDL criterion is computationally infeasible. Leclerc solved the problem by a continuation method, which he interpreted as adaptive smoothing of the image. At the beginning, the spatial scale of the smoothing filter is small, and within the iterations, the spatial scale increases except across the discontinuities, i.e. the regions boundaries. In 1994, Kanungo et al. [27] combined two-part coding with region merging so that the

derived MDL criterion was used to decide on merging the neighboring regions. A general view to region merging is given in Section 2.3.1. The MDL-based criterion derived by Kanungo was further developed in 2006 by Luo et al. [28] for image segmentation at multiple scales. The idea of segmentations at multiple scales is that the scale of the object is not known beforehand, or there are objects on multiple scales in the image and the decision of optimal scale can be dealt later in the analysis. Zhu et al. [36] combines the ideas of Leclerc's MDL criterion for segmentation, region merging, snakes, and circular windows to obtain a region competition algorithm. The algorithm minimizes the proposed energy function by alternating stages of optimizing region parameters, evolving the region boundaries and merging regions. Since all the stages are minimizing the proposed energy function, the iteration gives as a result a local minimum.

Our approach to MDL-based image segmentation is related to the approach of Kanungo et al. [27] and Luo et al. [28]. Therefore, their two-part coding based segmentation approach is next introduced in more detail. First, we will describe their two-part coding based region merging criterion. Then, their region merging algorithms are presented. After that, our approach is reviewed and experimental results of our algorithm on histological images are discussed in Section 4.6.

4.5.1 Two-part coding based MDL criterion for image segmentation

Kanungo et al. [27] proposed in 1994 an MDL-based image segmentation algorithm for multilayer images such as color images. The approach is inspired by two-part coding, which has already been introduced in Chapter 4.3.2. In the two-part coding based criterion, the total code length consists of two parts: the encoding of the model and the encoding of the data using the model. The total codelength to be minimized is

$$L(Y, M) = L(M) + L(Y|M), \tag{4.18}$$

where L denotes codelength, $L(M)$ codelength of the model, and $L(Y|M)$ the codelength of the data given the model. Therefore, the aim is to find the model M that minimizes the total codelength.

In Kanungo's approach [27], the data $Y = \{\mathbf{y}_i\}, i = 1, \ldots, n$ are colors of pixels of an image which are modeled such that the model, M, consists of two components: a segmentation, $\Omega = \{R_j\}$, $j = 1, \ldots N_R$, which partitions the image into N_R non-overlapping regions R_j; and the model parameters, $\beta = \{\beta_j\}$, $j = 1, \ldots N_R$ for modeling the pixel colors of each region separately. Hence, the total code length presented in Equation 4.18 can be written as

$$L(Y, \Omega, \beta) = L(\Omega) + L(\beta|\Omega) + L(Y|\Omega, \beta), \tag{4.19}$$

where $L(\Omega)$ is the codelength of the segmentation obtained by encoding the boundaries of the regions, $L(\beta|\Omega)$ is the codelength for encoding the model

parameters β given the segmentation Ω, and $L(Y|\Omega,\beta)$ is the codelength of the residuals given the boundaries Ω and the model parameters β. The encodings of each term of the total codelength are next introduced.

The codelength for encoding the segmentation, $L(\Omega)$, is obtained by encoding the boundary contours of the regions. The boundaries of the regions can be represented by a graph, where nodes are intersections of the boundaries and edges are branches of the boundaries between the intersections. Since the regions $\{R_j\}$ are assumed to be large, the cost of the graph is omitted and the cost for encoding the region boundaries consists only of the cost of encoding the branches. The paths of the branches along the image grid are encoded using chain codes in which each element of the chain represents the direction of the next step. Kanungo uses 4-connectivity in the image grid, which results that the required number of bits for the branch between regions R_i and R_j is

$$l_{ij}\log 3 + L^0(l_{ij}), \tag{4.20}$$

where l_{ij} is the length of the branch in terms of chain codes, $\log 3$ results from the three possible directions in the chain code, as the direction of last visited point is excluded, and $L^0(l_{ij}) = \log^*(l_{ij}) + \log c$ is Rissanen's universal prior for integers [90], where $\log^*(x) = \log x + \log\log x + \log\log\log x + \cdots$ up to all positive terms and $c = 2.865064$. Hence, the codelength for encoding the segmentation is

$$L(\Omega) = \sum_{ij}(l_{ij}\log 3 + L^0(l_{ij})). \tag{4.21}$$

Later in this thesis, two other approaches to encode the segmentation will be introduced. The approaches include the use of parameters of ellipses, which has been proposed in Publications II and III and will be presented in Section 4.6.1, and the Crack-edge-region-value (CERV) algorithm [91], which was originally designed to encode depth images but is also well suited for encoding segmentations, will be reviewed in Section 5.3.

The codelength for encoding the parameters given the regions is

$$L(\beta|\Omega) = \sum_j \frac{K_{\beta_j}}{2}\log n_j, \tag{4.22}$$

where K_{β_j} is the number of free parameters in β_j and n_j is the number of pixels in region R_j. The formula $(K/2)\log n$ is an asymptotic form (large n) of the optimal precision coding cost for encoding K independent real-valued parameters of a distribution that describe n data points [90].

The codelength for encoding the residuals given the segmentation and the parameters, $L(Y|\Omega,\beta)$, is obtained by a model in which the pixel colors $\mathbf{y}_i$ over each region are separately modeled by using polynomial grayscale surfaces, and the residuals are supposed to be spatially uncorrelated Gaussian distributed noise.

Hence, the parameters β_j of the model for pixel colors of the region R_j consists of polynomial coefficients of the grayscale surfaces $\boldsymbol{\mu}_j$ and the covariance matrix $\boldsymbol{\Sigma}_j$. The log likelihood for all the pixel colors Y_j belonging to the region R_j is given in [27] as

$$\log p(Y_j|\beta_j) = -\frac{n_j d}{2}\log(2\pi) - \frac{n_j}{2}\log|\boldsymbol{\Sigma}_j| - \frac{n_j d}{2}, \tag{4.23}$$

where d is the dimensionality of the color vector. Hence, the codelength for encoding the residuals given the segmentation and model parameters is

$$\begin{aligned} L(Y|\Omega,\beta) &= -\log p(Y|\Omega,\beta) = -\sum_j \log p(Y_j|\beta_j) \\ &= \sum_j \left[\frac{n_j d}{2}\log(2\pi) + \frac{n_j}{2}\log|\boldsymbol{\Sigma}_j| + \frac{n_j d}{2}\right]. \end{aligned} \tag{4.24}$$

The resulting total codelength to be minimized is

$$\begin{aligned} L(Y,\Omega,\beta) &= L(\Omega) + L(\beta|\Omega) + L(Y|\Omega,\beta) \\ &= \sum_{ij}(l_{ij}\log 3 + L^0(l_{ij})) + \sum_j \frac{K_{\beta_j}}{2}\log n_j \\ &\quad + \sum_j \frac{n_j}{2}\left[d\log(2\pi) + \log|\boldsymbol{\Sigma}_j| + d\right], \end{aligned} \tag{4.25}$$

where ij denotes indexing of the branches and j denotes indexing of the regions. In the following section, two approaches to minimize the total codelength will be introduced.

4.5.2 Region merging by using the two-part coding based MDL criterion

In the previous section, we reviewed a two-part coding based criterion for image segmentation proposed by Kanungo et al. [27]. This section introduces two approaches, Kanungo et al. [27] and Luo et al. [28], to minimize the criterion. Since obtaining the global minimum of the criterion is infeasible, both approaches propose segmentation algorithms that are based on region merging. Region merging has been introduced in general in Section 2.3.1.

In both approaches, when two regions R_i and R_j are merged into region R_k, the total codelength is decreased by

$$\Delta L(Y,\Omega,\beta) = \Delta L(\Omega) + \Delta L(\beta|\Omega) + \Delta L(Y|\Omega,\beta), \tag{4.26}$$

where

$$\begin{aligned} \Delta L(\Omega) &= l_{ij}\log 3 + L^0(l_{ij}) \\ \Delta L(\beta|\Omega) &= \frac{1}{2}\{K_{\beta_i}\log n_i + K_{\beta_j}\log n_j - K_{\beta_k}\log n_k\} \\ \Delta L(Y|\Omega,\beta) &= \frac{1}{2}\{n_i\log|\boldsymbol{\Sigma}_i| + n_j\log|\boldsymbol{\Sigma}_j| - n_k\log|\boldsymbol{\Sigma}_k|\} \end{aligned} \tag{4.27}$$

are the corresponding decreases in the coding lengths of segmentation $\Delta L(\Omega)$, parameters $\Delta L(\beta|\Omega)$ and fitting residuals $\Delta(Y|\Omega,\beta)$.

Kanungo's segmentation algorithm starts with polynomials having zero order. Once there are no more regions that would decrease the total codelength by merging, the algorithm tries the first order polynomials. This is continued until the algorithm converges, i.e. merging regions does not further decrease the total codelength.

Luo et al. [28] proposes a segmentation algorithm which produces multiscale segmentations and representations. The motivation for the multiscale segmentation stems from the idea that objects and structures are dependent on the scale of the observation. Therefore, the aim of the multiscale segmentation is to generate segmentation results in multiple scales so that the following image analysis steps such as object detection can make the decision of the optimal scale. One way of obtaining multiscale representations is to use low-pass filters at different scales to smooth the image. The problem of using smoothing is that it does not take into account local structures and important region boundaries may be distorted. Luo et al. [28] perform smoothing within each region and it thus adapts to local structures. The smoothing transformation $\mathbf{T}$ for a pixel vector $\mathbf{y}_i$ in a region R_j having parameters $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ is defined as

$$\mathbf{T}(\mathbf{y}_i) = (\mathbf{y}_i - \boldsymbol{\mu}_j)|\boldsymbol{\Sigma}_j|^{(\lambda-1)/2d} + \boldsymbol{\mu}_j, \tag{4.28}$$

where λ is the scale parameter. The correspondence between the covariance matrices before $\boldsymbol{\Sigma}_j$ and after $\boldsymbol{\Sigma}'_j$ the transformation is

$$\log(|\boldsymbol{\Sigma}'_j|) = \lambda \log(|\boldsymbol{\Sigma}_j|). \tag{4.29}$$

Since the smoothing affects only the codelengths of the residuals, it is decreased by

$$\Delta L'(Y|\Omega,\beta) = \lambda \Delta L(Y|\Omega,\beta), \tag{4.30}$$

and the decrease in the total codelength after the transformation is

$$\Delta L'(Y,\Omega,\beta) = \Delta L(\Omega) + \Delta L(\beta|\Omega) + \lambda \Delta L(Y|\Omega,\beta). \tag{4.31}$$

In addition, Luo's segmentation algorithm [28] applies mean shift clustering [44], discussed in Section 2.3.3, to obtain an over-segmentation which is followed by region merging at multiple scales. The aim of the over-segmentation is to produce reasonable region sizes for the following region merging, so that the computation of covariances and the region boundaries would be more reliable than in the case of starting with one-pixel regions.

4.6 Ranking among competing interpretations of a clump by using the MDL principle

Publications II and III present an MDL-based approach for ranking different interpretations of cell nuclei clumps. The approach interprets cell nuclei by ellipses, such that each ellipse represents one nucleus. The proposed approach is similar to Kanungo et al. [27], which has been reviewed in Section 4.5.1, but compared to Kanungo's approach, our approach uses implementable descriptions and codelengths, and does not require any asymptotic approximations. In addition, we have represented the segmentation regions by a union of ellipses instead of using chain codes. The direct optimization of the proposed criterion is difficult, and some preliminary clump splitting results are necessary. The SNEF algorithm proposed in Publication I, and reviewed in Section 3.4, efficiently fits ellipses to cell nuclei clumps in H&E stained histological images and gives satisfactory results in clear cases. One of the disadvantages of the SNEF algorithm for general use is related to the thresholding methods applied within the algorithm, since by changing the thresholding methods or working on histological images that contain highly heterogeneous cell nuclei, the results may be different. The proposed MDL-based criterion addresses the need of improving the SNEF algorithm and provides a more sophisticated approach to select between competing interpretations of a clump of cell nuclei. We have studied the proposed MDL-based approach two ways: by producing different interpretations for cell clumps by varying thresholding methods in the SNEF algorithm, and by spatial transformations induced to the original image. In the experiments, the clump splitting results are compared to the ground truth ellipses formed from splitting results provided by several human subjects. In addition, a simple local method to improve ellipse parameters is presented.

Next, the proposed MDL-based criterion used for ranking different image interpretations is described. Then, we will introduce the modifications we made to the SNEF algorithm for obtaining a number of different image interpretations. In addition, we will review the two spatial transformations we made for the image before applying the SNEF algorithm. After that we will describe the formation of the ground truth ellipses applied in the experiments. Finally, a summary of the experimental results is given.

4.6.1 An implementable MDL-based description of the image with unions of ellipses

The inspiration for the MDL-based description of the image, presented in Publications II and III, is given by Kanungo et al. [27] and Luo et al. [28]. Their approaches are described in Sections 4.5.1 and 4.5.2. The main idea which we adopted from those papers is the cost of losslessly encoding an image by encoding the segmentation first, and then encoding the image with help from the segmenta-

tion. Therefore, the total code length, $L(Y, \Omega, \beta)$, takes the same form as the total code length proposed by Kanungo, and also presented in Equation 4.19, being

$$L(Y, \Omega, \beta) = L(\Omega) + L(\beta|\Omega) + L(Y|\Omega, \beta), \tag{4.32}$$

where $L(\Omega)$ is the codelength for encoding the boundaries of the regions, $L(\beta|\Omega)$ is the codelength for encoding the model parameters β given the boundaries of the regions Ω, and $L(Y|\Omega, \beta)$ is the codelength for the residuals given the boundaries Ω and model parameters β.

In our approach, the implementation of the segmentation, models, and parameters are different. Namely, our approach is based on an implementable and non-asymptotic two-part code of MDL. In addition, the segmentation regions are interpreted by possibly overlapping ellipses, instead of Kanungo's chain codes for generic regions. The union of ellipses define the foreground region, and the remaining image pixels form the background. The foreground and background are encoded separately using different coding parameters. We assume a constant model within a region, and residuals are encoded using Golomb-Rice coding [92], reviewed in Section 5.1.3.2. Next, the three terms of the total codelength of the proposed approach are introduced in more detail.

The first term, the codelength for encoding the boundaries of the regions, $L(\Omega)$, is as follows. In the proposed approach, the regions are formed by a number of n_E ellipses such that for each ellipse, pixels inside the contour of the ellipse are denoted as $\mathcal{E}_i$, and the union of the points inside the ellipse contours $\Omega_F = \cup_{i_1}^{n_E} \mathcal{E}_i$ defines the region of a clump, or the foreground. The background, Ω_B, is formed by the remaining pixels of the image. Each ellipse, $\mathcal{E}_i$, is described by five ellipse parameters $[x_{0_i}\ y_{0_i}\ a_i\ b_i\ \theta_i]$, where (x_{0_i}, y_{0_i}) are the coordinates of the center point of the ellipse, a_i and b_i are the lengths of the major and minor axes, respectively, and θ_i is the angle between the major axis and x-axis. Different parametrizations of ellipses are described in Section 3.3.1. The parameters of ellipses are encoded such that each parameter is uniformly quantized using same number of bits, b, to the range of possible parameter values which are from 0 to following maximum values $[n_r\ n_c\ \sqrt{n_c^2 + n_r^2}\ \sqrt{n_c^2 + n_r^2}\ \pi]$, where n_c and n_r are number of columns and rows in the image, respectively. As a result, the codelength for representing the ellipses is $L(\Omega) = 5bn_E$.

The second term, the codelength for encoding the model parameters β given the boundaries of the regions Ω, $L(\beta|\Omega)$, is next described. We use a constant model for the foreground and background separately such that average values of the foreground μ_F, and of the background μ_B are both transmitted using 8 bits for each. Therefore, the codelength for the model parameters is $L(\beta|\Omega) = 16$ bits.

The third term, the codelength for the residuals given the boundaries Ω and model parameters β, $L(Y|\Omega, \beta)$, is as follows. The residual of a pixel (x, y) is $Y(i, j) - \mu_F$, if $(x, y) \in \Omega_F$, and $Y(i, j) - \mu_G$, if $(x, y) \in \Omega_B$. We assume that the residuals

are from exponentially decaying probability distribution for which Golomb-Rice codes are optimal Huffman codes. In addition, the Golomb-Rice codes are known to be efficient among lossless image compression applications. The Golomb-Rice codes are reviewed later in Chapter 5.1.3.2, where we discuss more about lossless image compression. Here, we apply Golomb-Rice coding with different Golomb-Rice parameters l_F and l_B for the residuals of the foreground and of the background, respectively. The optimal Golomb-Rice parameters are transmitted before encoding of the residuals using 8 bits each. Then, negative residual values are mapped to non-negative ones γ_i. The encoding of the number of n_F foreground pixels by Golomb-Rice coding costs altogether $L_F = n_F(l_F + 1) + \sum_{i=1}^{n_F} \lfloor \frac{\gamma_i}{2^{l_F}} \rfloor$. The cost for encoding the residuals of the background L_B is obtained similarly. Hence, the total codelength for residuals is $L(Y|\Omega, \beta) = L_F + L_B + 16$ bits.

4.6.2 Different image interpretations by varying the thresholding methods of the SNEF algorithm

The direct optimization of the proposed MDL-based description of the image is time-consuming and very inefficient, since the increasing number of ellipses increases the number of parameters to be estimated, and as discussed in Section 3.3.2, ellipse fitting is a nonlinear optimization problem. The SNEF algorithm, proposed in Publication I and reviewed in Section 3.4, is designed to give fast and efficient clump splitting results for histological images by fitting ellipses to the various pixel sets derived from the edge pixel image. The main concerns are related to the thresholding methods within the algorithm, since deciding a proper thresholding method that would work on all images is difficult. A general introduction to image thresholding methods is given in Section 2.1.

The SNEF algorithm is selected to provide different segmentations for the ranking with MDL-based criterion by varying the thresholding methods in a couple of ways. The first place, where thresholding is used in the SNEF algorithm, is finding the initial locations of the clumps by thresholding the gray level image. Originally, the SNEF algorithm applies dual thresholding [32]. The motivation for using dual thresholding stems from histological images having three different tissue components: cell nuclei, cytoplasm, and background. However, sometimes the background is missing, and a better way is given by Otsu's method [31], which is a thresholding method for two classes, and is therefore selected to be the other option for finding the initial locations of the clumps.

The second place, where thresholding is applied in the SNEF algorithm, is the thresholding of the gradient magnitude image with the idea that those segments resulting from the thresholding would help in separating the touching nuclei into individuals. To obtain the thresholded gradient magnitude image, we use two thresholds: one resulting from Otsu thresholding, and the other threshold value being zero, which completely ignores the gradient magnitude image. The reason for the possibility of ignoring the gradient magnitude is that in highly textured

scenes, the Otsu's method might result in numerous segments inside nuclei, which prevents the rotating ray from entering the real borders of nuclei, and the idea of gradients guiding the splitting is then lost. With these aforementioned additions to the thresholdings of the SNEF algorithm, we result in altogether four different combinations and possibly four different clump interpretations.

4.6.3 MDL variability under two spatial transformations

In Publication III, we have studied the variability of the proposed MDL-based criterion by two spatial transformations: smoothing and down-scaling. These spatial transformations are applied to the original image before the execution of the SNEF algorithm. The idea behind using the spatial transformations is that they reduce the level of noise in the image, blur unwanted patterns such as chromatin texture within nuclei which may be mixed with real edges, and speed-up the segmentation process. The effects of the spatial transformation are evaluated in two ways: by comparing the resulting MDL values and the execution times. Next, we specify the applied smoothing and down-scaling transformations after which we describe how the different images are made comparable for the evaluation of the results.

Smoothing can be used as a pre-processing step for the segmentation algorithms to speed-up and possibly to increase robustness of the following segmentation process. The difficulty of applying the smoothing to segmentation is to select the level and range of smoothing, since the smoothing may distort boundaries that are important for the segmentation. In our experiments, we used Gaussian filtering in the 3-by-3 neighborhood with five standard deviations $\sigma_i \in \{0.2;\ 0.4;\ 0.6;\ 0.8;\ 1.0\}$. The bigger the value of the standard deviation of the Gaussian filter, the more smoothing effect is applied to the original image. The Gaussian kernel that is convolved with the image is

$$K(x,y) = C^{-1} e^{\frac{x^2+y^2}{2\sigma^2}}, \tag{4.33}$$

where $x, y \in \{-1, 0, 1\}$, and the normalization constant, C, is

$$C = \sum_{j\in\{-1,0,1\}} \sum_{k\in\{-1,0,1\}} e^{\frac{j^2+k^2}{2\sigma^2}}. \tag{4.34}$$

Down-scaling, reduces the number of pixels within an image. Hence, it most likely simplifies computations and accelerates the segmentation and clump splitting procedures. The disadvantages of down-scaling are that not all the information from the original image is available and the results may become rough as the level of down-scaling increases. In the experiments, we scaled the original images by five scaling factors, $s_i \in \{0.9;\ 0.8;\ 0.7;\ 0.6;\ 0.5\}$, so that the scaling factor corresponds to the factor of decrease of the original dimensions. The pixel values

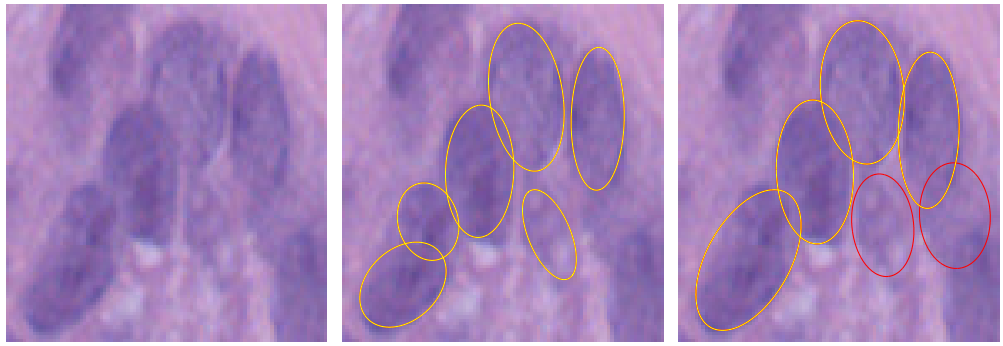
for the scaled images are obtained by bicubic interpolation which outputs the weighted average of the 4-by-4 neighborhood of a pixel [93, 94].

Once we have obtained the SNEF results, the resulting interpretations need to be made comparable among different sized images. For that purpose, we utilize relative measures so that for example in case of interpretations resulting from down-scaled images, the ellipse parameters are mapped to the original image size and the evaluation of the MDL-based criterion is done on the original images. In addition, the resulting values of the MDL criterion are divided by the value of the MDL criterion resulting from the interpretation for which no spatial transformations is applied. This results in the relative values of unprocessed images equaling to 1. If the relative MDL value after spatial transformation results in a value less than one, it means that the spatial transformation has utilized the SNEF algorithm to find a better interpretation. The relative execution times are computed similarly, i.e. each execution time is divided by the execution time of the SNEF algorithm on the original image.

4.6.4 Segmentation evaluation against the sets of ground truth ellipses

Segmentation results can be evaluated in multiple ways. One way is to compare the obtained results to the ground truth. Typically, the ground truth is a single segmentation or interpretation for a clump and the comparison can be, for instance, the comparison between the numbers of segmentation regions or the comparison between the pixel-wise agreements. However, the ground truth may also be a consensus or combination of several interpretations, as in Publications II and III. Next, the formation of the ground truth interpretations used in Publications II and III is described, after which two different ways of comparing the results against the ground truth are presented.

In Publications II and III, the experiments are performed on histological images, and due to the complex nature of histological images, as discussed in Section 3.1, there can be multiple opinions for the correct number of ellipses and their locations. Therefore, the ground truth interpretations are based on a database provided by human subjects, where each subject was able to give several interpretations for an image. The subjects' interpretations are obtained via an interactive graphical routine which is as follows. First, each subject marked numerous possible ellipse resembling objects in an image by ellipses. The subjects were advised to include only ellipses that are fully covered in the image. A single object is obtained by marking initial border pixels of the object, after which an ellipse is fitted to the marked pixels and shown to the subject over the original image. The subject is able to adjust the ellipse to better fit his opinion of the borders of the object. Once the set of ellipses are traced, the subject forms interpretations by combining the ellipses into most preferred configurations, and gives his subjective opinion of the degree of belief or confidence for each of his interpretations. In Figure 4.3, all the



(a) **(b)** **(c)**

Figure 4.3: Ellipses traced by two subjects, and their highest confidence interpretations. (a) Original image. (b) The subject S_1 has given only one interpretation and its ellipses (on yellow). (c) The ellipses of the subject S_2 highest confidence interpretation (on yellow), and the ellipses used in alternative interpretations (on red).

ellipses traced by two subjects, and the subjects' highest confidence interpretations for an image are shown. It can be seen that the second subject also traced ellipses that were not included to the highest confidence interpretation.

The first approach to evaluate segmentation or interpretation against the ground truth is based on comparing quantities, such as number of ellipses and the values of the proposed MDL criterion. The comparison is done by measuring the difference of the algorithm given quantity in question to the subjects' average value of the quantity, and evaluating the size of the difference by the standard deviation of the subjects' interpretations. For instance, the average number of ellipses for image I_i is

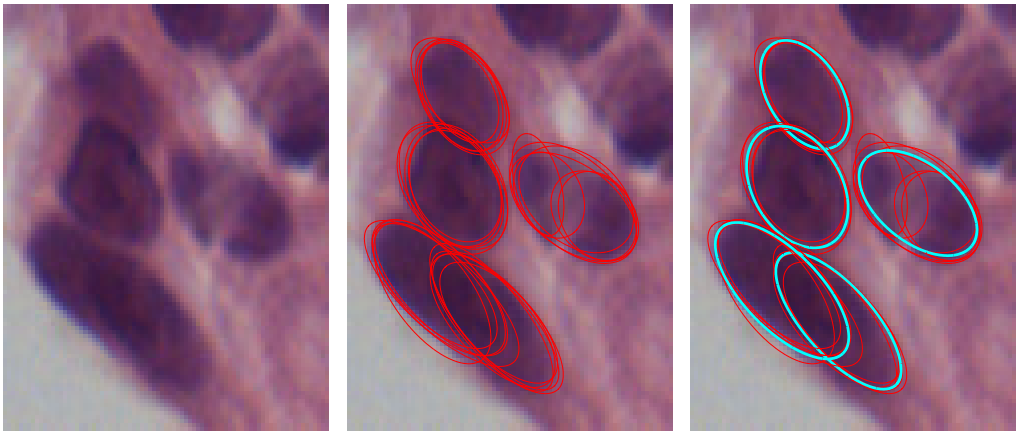
$$\tilde{n}_E(I_i) = \frac{1}{n_S} \sum_{k=1}^{n_S} \sum_{\ell=1}^{n_C(k)} \hat{p}(I_i, S_k, C_\ell) n_E(I_i, S_k, C_\ell), \tag{4.35}$$

where $n_S = 5$ is the number of subjects; S_k is a subject and $n_C(k)$ is the number of interpretations the subject S_k has given for the image I_i; $n_E(I_i, S_k, C_\ell)$ is the number of ellipses in the subject's S_k interpretation C_ℓ, and the $\hat{p}(I_i, S_k, C_\ell)$ is the degree of confidence of the interpretation. The variance of the number of the ellipses for an image I_i is

$$\sigma^2(n_E(I_i)) = \frac{1}{n_S} \sum_{k=1}^{n_S} \sum_{\ell=1}^{n_C(k)} \hat{p}(I_i, S_k, C_\ell) \left[n_E(I_i, S_k, C_\ell) - \tilde{n}_E(I_i) \right]^2 . \tag{4.36}$$

The difference of the obtained interpretation result C_R from the average is measured as

$$\Delta(n_E(I_i)) = |n_E(I_i, C_R) - \tilde{n}_E(I_i)|. \tag{4.37}$$



(a) **(b)** **(c)**

Figure 4.4: Obtaining ground truth ellipses by averaging the subjects' ellipses. (a) Original image. (b) All ellipses traced by the 5 human subjects (thin red). (c) The ground truth ellipses (bold cyan) obtained by averaging the subjects' ellipses.

Similarly, the average and variance quantities can be derived for the subjects' MDL values, and the difference of the obtained MDL value and the average value can be compared to the standard deviation.

The second approach to compare the obtained results with the ground truth is based on pixel-wise agreement of two segmentations or clump splitting results. Therefore, the ground truth for an image is a single segmentation result. In Publication III, the ground truth segmentation for an image is formed from several segmentations such that ellipses are grouped into clusters where each cluster represents one nucleus after which each cluster is averaged by fitting an ellipse to the contour pixels of the ellipses of a cluster. These averaged ellipses form the set of ground truth ellipses. During the process, we discarded some small degree of confidence ellipses, which were part of the alternative low confidence interpretations given by some subjects, or ellipses that were in an area only a minority of the subjects considered to contain an object. At the end, the ground truth ellipses were validated by an expert pathologist. In Figure 4.4, all ellipses traced by the 5 human subjects for an image, and the ground truth ellipses obtained by averaging the subjects' ellipses are shown.

To measure the pixel-wise agreement of two segmentations or interpretations, we have applied two measures: the first is directly based on popular dice coefficients [95], and the second is a modified version of it, designed specifically for the comparison of two clump splitting results. The first measure is following. The dice coefficient for two sets of foreground pixels A and B is given as

$$D_A = \frac{2|A \cap B|}{|A| + |B|}, \tag{4.38}$$

where $|A|$ and $|B|$ denotes the cardinality of a set A and of a set B, respectively. A value of 1 indicates that the sets are perfectly overlapping and a value of 0 indicates there is no overlap.

In case of splitting objects, and a segmentation consisting of a foreground formed from several objects, such that $A = \bigcup_{i=1}^{n} A_i$ and $B = \bigcup_{j=1}^{m} B_j$, one might want to use a refined version of the original dice coefficients. In Publication III, the second used pixel-wise measure is as follows. First, the reference segmentation is selected, since the number of regions can be different in different interpretations. The reference segmentation is selected to be the one that has a higher number of regions, which means that if $n \geq m$ holds, then A is the reference segmentation. Then, for each region A_i the largest intersecting region from the set B is selected and denoted as $B_{j(i)}$. The pairwise similarities are

$$D_{A_i,B_{j(i)}} = \frac{2|A_i \cap B_{j(i)}|}{|A_i| + |B_{j(i)}|} \tag{4.39}$$

and the measure for the pixel-wise agreement of two interpretations is the average of the pairwise similarities, which is denoted as

$$D_R = \frac{1}{n_A} \sum_{i=1}^{n_A} D_{A_i,B_{j(i)}}. \tag{4.40}$$

4.6.5 Data set and experimental results

The experiments are performed over a data set of H&E stained histological images. The tissue samples for the images are taken from the epithelium of the lower part of esophagus suspected of having Barrett's esophagus, an abnormal change in tissue structure in which the tissue lining of esophagus is replaced by the lining of intestine [96]. There is a small risk that the Barrett's esophagus develops into invasive adenocarcinoma. The time of progression to invasive adenocarcinoma is mainly predicted by the degree of dysplasia, abnormally growing cells in Barrett's esophagus. Currently, the grading of dysplasia, and the probability of developing the cancer, is mostly based on subjective opinion that relies on the evaluation of the nuclear characteristics in histological images.

In our experiments, we used a data set that contains different grades of epithelial dysplasia in Barrett's esophagus. There are altogether 24 images and the sizes of the images varies based on the size of the cell nuclei clump. Each image contain one clump, and the numbers of nuclei within the clumps vary. The ground truth ellipses for the images are specified in Section 4.6.4. Shortly, the ground truth ellipses consist of 5 subjects' opinions for the locations of the ellipse resembling objects, i.e. cell nuclei. The subjects were allowed to give for an image several interpretations consisting of different ellipse combinations. The subjects were also asked to give their opinion of how likely they think that combination is. The

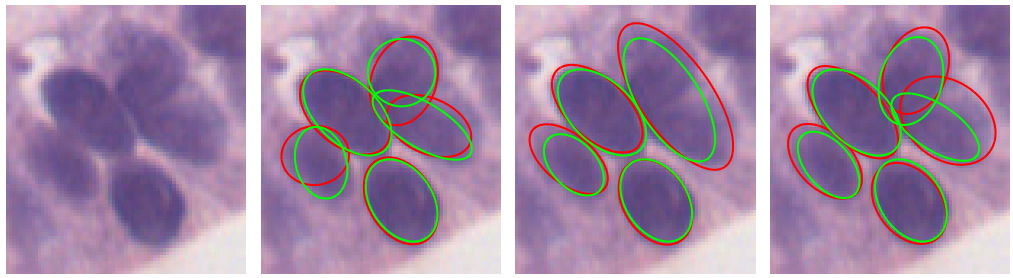
segmentation results were also verified by an expert pathologist specialized to the Barrett's esophagus.

We have performed two different experiments. Both experiments aim to study the proposed MDL-based criterion in ranking competing interpretations of a clump. The first experiment is originally presented in Publication II. The different interpretations for ranking are obtained by varying thresholding methods of the SNEF algorithm (see Section 4.6.2). In addition, the ellipse parameters resulting from the SNEF algorithm are improved by a simple local iterative algorithm. In the second experiment, presented in Publication III, the various interpretations for ranking are provided by smoothing and scaling the original image before applying the SNEF algorithm (see Section 4.6.3). In addition, the second experiment provides an important evaluation of the agreement of the best (lowest) MDL values and the best (highest) pixel-wise similarity coefficients.

In the first experiment, originally presented in Publication II, the MDL values of the subject provided interpretations are compared to the MDL values of the interpretations provided by varying the thresholding methods of the SNEF algorithm specified above in Section 4.6.2, and using measures described in Section 4.6.4 and presented in Equations 4.35-4.37. In addition, both the subjects' interpretations and the interpretation of the SNEF algorithm corresponding to the lowest MDL value are optimized by a simple local iterative algorithm.

The proposed simple iterative algorithm for optimizing the MDL criterion is as follows. At each iteration, parameters of one ellipse are studied and possibly changed while the other ellipses are kept fixed. The parameters of the studied ellipse are varied such that small changes, λ, are induced to all of the five ellipse parameters. For each ellipse parameter α, there are altogether 3 different possibilities: $\alpha - \lambda$, α, and $\alpha + \lambda$. As a result, there are 3^5 different parameter combinations for an ellipse. The parameter combination resulting in the lowest total codelength of the MDL criterion is selected. Once the parameters of the ellipse are updated towards the lower total codelength of the MDL criterion, the iterative algorithm updates all the other ellipses one by one in similar manner. Once we have gone through all the n_E ellipses, one relaxation cycle of the iterative algorithm ends. In the experiments, there were 5 relaxation cycles with the values of λ being from the list $[1, 2, 1, 2, 1]$. The effect of the algorithm to the ellipses is visualized in Figure 4.5.

It turned out that the proposed local iterative algorithm reduces the variance of the MDL criterion over the provided human interpretations, as shown in Figure 4.6. In addition, the difference of the MDL values of the SNEF algorithm from the ground truth are in general two times lower than the standard deviations of the human subject obtained MDL. The differences are even smaller after the iterative algorithm, about one standard deviation. The differences in the number of ellipses of the SNEF algorithm from the ground truth is about the level of standard deviation of the variability of the human provided number of ellipses.



(a) (b) (c) (d)

Figure 4.5: The effect of the local iterative algorithm to the ellipses. The initial ellipses are shown in red, and the ellipses after the algorithm in green. (a) Original RGB image. (b) An interpretation provided by the SNEF algorithm. (c) An interpretation provided by a subject. (d) Another interpretation by a subject. The figures are originally published in Publication II, first published in the Proceedings of the 19th European Signal Processing Conference (EUSIPCO-2011) in 2011, published by EURASIP.

In the second experiment, originally presented in Publication III, the spatial transformations, smoothing and scaling, are applied to the original image before applying the SNEF algorithm. The experimental settings of the smoothing and scaling are explained in Section 4.6.3. In Publication III, it is shown that increasing levels of down-scaling decrease the execution times of the SNEF algorithm, such that with scaling factor 0.5 the execution times are on average less than 0.25 of the execution times at the original image size. The disadvantage of the down-scaling is that it on average increases the MDL values, although for some images there are improvements in the MDL values. In the smoothing experiment, there are on average no significant improvements or deterioration with the MDL values or execution times. On the other hand, in about half of the cases, the obtained MDL values are better than those of the original image. Therefore, if time allows, one could run the SNEF algorithm twice, once with original image, and again with smoothed image, and choose the interpretation that gives a lower MDL value.

In Publication III, the clump splitting results are also evaluated against the ground truth interpretations by two dice-coefficient-based [95] similarity indexes: an area-wise index, S_A, and en ellipse-wise index, $S_{\mathcal{E}}$, as described in Section 4.6.4, and presented in Equations 4.38 and 4.40, respectively. The agreement between the similarity indexes and the MDL values of the interpretations are studied over 24 images. For each image, the SNEF algorithm provided after spatial transformations to the original image 11 possibly different interpretations, and human subjects provided 5 interpretations, since we considered from each subject only the most likely labeled interpretation. In the ideal case, the interpretations of the lowest MDL values are giving the highest similarity index values so that the proposed MDL based criterion could be used to rank different interpretations, and to select the most optimal interpretation.

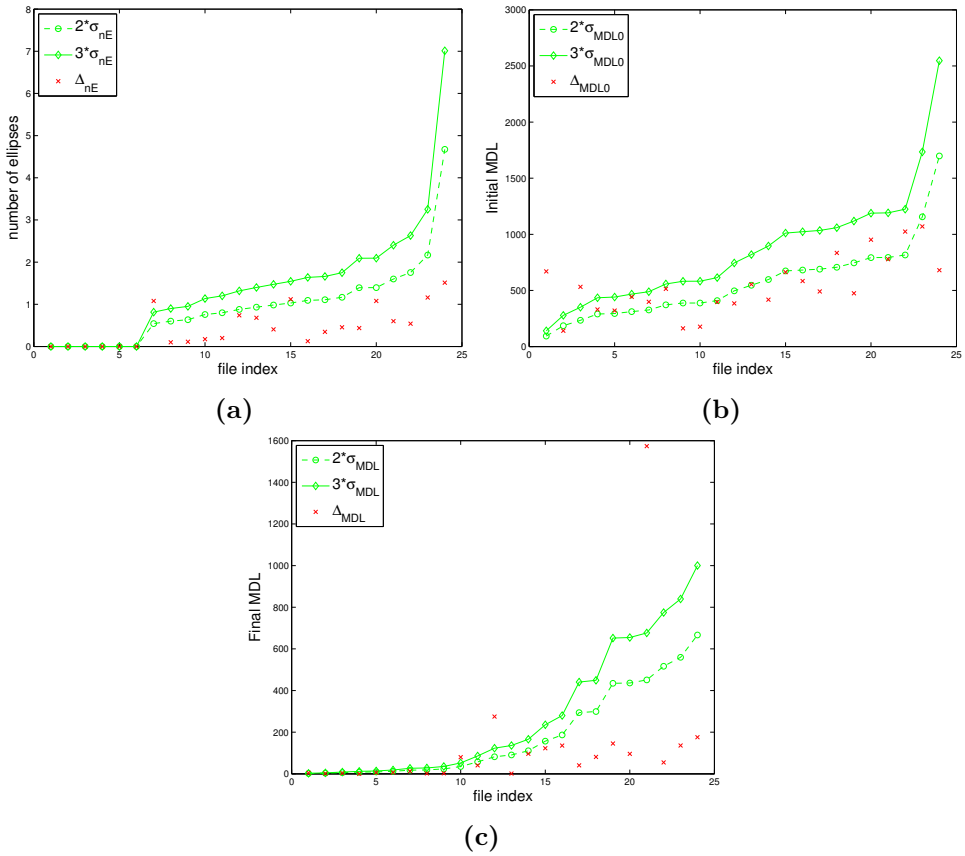


Figure 4.6: The variabilities of the subjects' interpretations (in green) from the estimated average quantities measured on the data set of 24 images. The used quantities are: (a) the number of ellipses, (b) the MDL criterion before the iterative algorithm, and (c) the MDL criterion after the iterative algorithm. The deviations of the best SNEF results (based on the MDL criterion) from the estimated averages are shown on red points. The figures are originally published in Publication II, first published in the Proceedings of the 19th European Signal Processing Conference (EUSIPCO-2011) in 2011, published by EURASIP.

From the results shown in Figure 4.7, it can be seen that the area-wise similarity indexes S_A of the lowest MDL value giving interpretations are the same or close to the values of the highest similarity indexes S_A, especially with the SNEF algorithm provided interpretations. In addition, the S_A values of the lowest MDL value giving SNEF interpretations are in most cases close to the subjects' highest S_A values. With the ellipse-wise similarity index $S_\mathcal{E}$, the results are similar, but with one major difference: in some images, the $S_\mathcal{E}$ values of the lowest MDL interpretations are far away from the highest obtained $S_\mathcal{E}$ values. In those images where the values of area-wise S_A are close to the highest values, it means that the ellipses of the MDL optimal interpretation are located near the ground truth, but the number of ellipses is different from the ground truth. The average (means

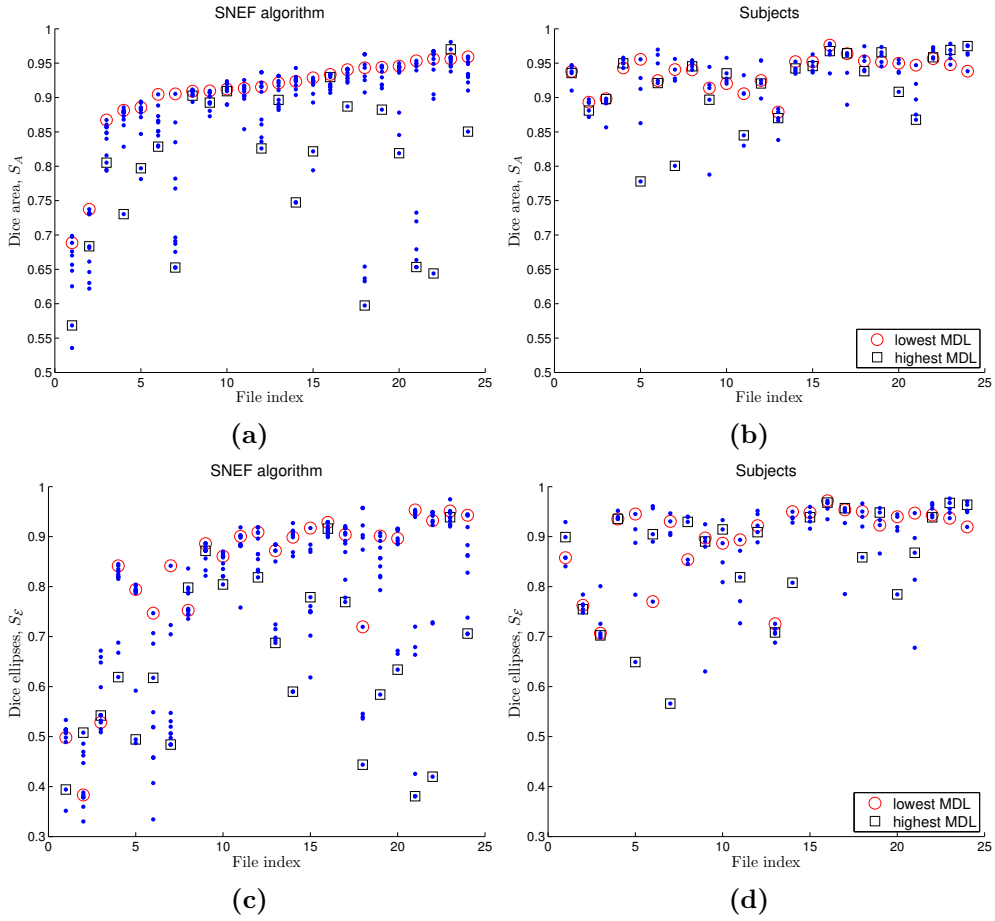


Figure 4.7: Evaluation of the ellipse interpretations against the ground truth ellipses over the data set of 24 histological images. Two similarity indexes have been used: the area-wise, S_A, and ellipse-wise, $S_\mathcal{E}$, similarity indexes. The area-wise similarity index is used in images (a) and (b), and the ellipse-wise in images (c) and (d). In images (a) and (c), the interpretations for each of the 24 image files consist of 11 interpretations provided by the SNEF algorithm after the spatial transformation experiments. In images (b) and (d), there are 5 subjects' interpretations for showing the variability among the segmentations provided by the subjects. The value of the similarity index for each interpretation is shown by a blue dot. The higher the similarity index is, the better the interpretation is agreeing with the ground truth. We have also marked the interpretations that gives the lowest (red ring) and the highest (black squares) values of the MDL criterion. In the optimal case, the highest similarity index values are given by interpretations that have the lowest values of the MDL criterion (red ring). The figures are originally published in Publication III. Reprinted with permission from Springer ©2013.

and medians) values of the area-wise S_A and ellipse-wise $S_\mathcal{E}$ similarity indexes of the interpretations corresponding to the lowest MDL values presented separately for the SNEF algorithm and subjects are shown in Table 4.2. It can be seen

Table 4.2: The averages (*mean* / *median*) of the area-wise S_A and ellipse-wise $S_\mathcal{E}$ similarity indexes of the interpretations corresponding to the lowest MDL values on each of the 24 images. The lowest MDL value corresponding interpretation for each image is selected from the set of 11 interpretations in the case of the SNEF algorithm, and from 5 interpretations in the case of the subjects.

	SNEF	Subjects
S_A	0.9057 / 0.9182	0.9361 / 0.9419
$S_\mathcal{E}$	0.8234 / 0.8912	0.8946 / 0.9267

that the average values resulting from the SNEF algorithm with MDL ranking are quite close to subjects' average values. The failure of the MDL ranking with SNEF algorithm interpretations on some images is seen on mean values of the ellipse-wise similarity index $S_\mathcal{E}$. However, since the number of failed images is small, the median value of the MDL optimal similarity index $S_\mathcal{E}$ is 0.891 and close to values obtained from subject interpretations.

The difference between the results of the interpretations provided by the SNEF algorithm and by human subjects is that on some images, the subjects' highest MDL value interpretation gives higher similarity index values than the lowest MDL value interpretation (see Figure 4.7). It might seem that the proposed MDL criterion has failed to select between different interpretations. However, this phenomenon results from the fact that even small shifts from the optimal ellipse locations may increase the value of the MDL criterion considerably. Therefore, interpretations even close to the ground truth may result into high MDL criterion values.

4.6.6 Conclusions of the chapter

This chapter described methods for ranking segmentations by using the MDL principle. We noticed that the lowest MDL values and the highest values of the similarity indexes were correlating well. Therefore, the results suggest that the proposed MDL-based criterion is applicable to select an interpretation for a clump of cell nuclei which is close to ground truth.

The segmentations are ranked by the total codelengths of encoding the segmentation and encoding the image conditional on the segmentation. However, the total codelengths do not always result in the lowest possible codelengths for an image. Better codelengths can be obtained by linear predictive lossless image compression approaches which efficiently take into account spatial correlations often present in images. On the other hand, lossless image compression algorithms applicability to rank segmentations and interpretations is problematic, as will be discussed within the next chapter.

5 Using medical image segmentation for lossless compression

The aim of image compression is to describe an image using a smaller amount of bits so that it can be stored and transmitted more efficiently. In lossless compression, no information is lost; and hence, the original image is fully recoverable from the compressed image. Although lossy compression methods can achieve much higher compression ratios than lossless compression methods, it is often essential to avoid artifacts and loss of information induced by lossy compression. One important set of images for lossless compression are medical images, where no information is allowed to be lost on diagnostically important regions, since artifacts resulting from lossy compression may lead to false conclusion.

In this chapter, Publications V and VI are reviewed. Both publications present several linear predictive lossless image compression algorithms for medical images, more precisely histological (in Publication V) and retinal color images (in Publication VI). The main goal of the algorithms has been adding segmentation to the lossless image compression. Therefore, the algorithms are encoding images in two phases such that an image segmentation is encoded first, after which the residual image, the difference between the original image and the segmentation image, is transmitted. Some of the proposed compression algorithms are encoding image segmentation regions independently. Hence, these algorithms enable lossless encoding to be used only in the regions-of-interest, and allows lossy coding in other parts. The other reason to include image segmentation to the lossless image compression algorithms stems from the previous chapter, where we presented an information theoretical approach to image segmentation. The different image segmentations were ranked based on their total codelengths resulting from the encoding of a segmentation, and the encoding of the image conditional on the segmentation. The MDL-inspired image segmentation approaches do not take into account spatial dependencies of pixel values, which often occur in images and are modeled in lossless image compression algorithms via causal prediction templates. In this chapter, we will discuss the applicability of the context dependent linear

predictive lossless image compressors in ranking competing image segmentations.

The differences between the proposed algorithms are mainly due to prediction templates, and on how the algorithms utilize contexts. In Publication VI, we have studied three sparse predictor design methods to select from the causal prediction templates the most significant template elements. The sparse predictor design method and the sparsity level are selected for the final experiments by an approach inspired by the MDL principle.

The rest of this chapter is as follows. First, an introduction to predictive lossless image compression is given. Then, four publicly available lossless image compressors, which include CALIC [13], LOCO-I [14], JPEG 2000 [97, 98] and LCIC [99], are introduced. After that, the lossless compression algorithm applied in Publications V and VI to encode segmentation images is reviewed. Finally, the proposed lossless compression algorithms are presented, and the experimental results are summarized.

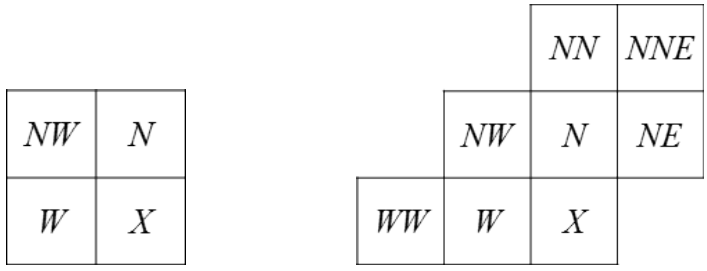
5.1 Introduction to predictive lossless image compression algorithms

Two famous lossless compression algorithms for gray level images, CALIC [13] and LOCO-I [14], apply prediction and context modeling for efficiently predicting the image pixel values. Our lossless compression schemes presented in Publications V and VI have similar grounds. Therefore, a general overview of predictive, context-based lossless compression will be next described. The presented concepts include prediction in image compression, context coding, and residual coding with two popular approaches: arithmetic coding and Golomb-Rice coding.

5.1.1 Prediction in lossless image compression

In images, values of pixels are usually spatially correlated such that each pixel is similar to or dependent on neighboring pixels. Pixels are scanned in some pre-defined order which may be line-by-line, column-by-column, or zig-zag. The aim of the prediction is to guess the value of current pixel x_{i+1} based on a subset of the available past sequence x^i, called a causal template.

Naturally, templates and prediction methods differ from one algorithm to another. For instance, in LOCO-I, the template consists of only three neighboring pixels: N is north from current pixel, W is west, and NW is northwest, respectively. In Figure 5.1(a), the corresponding pixel locations of the template for predicting the pixel X is shown. Due to the line-by-line pixel scanning order, N, W, and NW are available to both encoder and decoder. The prediction is done according to a variation of median adaptive prediction [100] so that the prediction for the



(a) LOCO-I **(b)** CALIC

Figure 5.1: Templates for predicting the pixel X in lossless predictive image compressors: (a) LOCO-I, and (b) CALIC.

current pixel X is given by

$$\hat{X} = \begin{cases} \min(N, W) & \text{if } NW \geq \max(N, W) \\ \max(N, W) & \text{if } NW \leq \min(N, W) \\ N + W - NW & \text{otherwise} \end{cases} . \tag{5.1}$$

Therefore, the predictor in LOCO-I can be seen as a test to detect vertical and horizontal edges; and in case of smooth area, the prediction will be $N + W - NW$.

In CALIC, the template is wider and the prediction is more complex. The template for prediction consists of pixels N, W, NW, NE, NN, WW, and NNE; and the template is shown in Figure 5.1(b). The prediction is based on gradient-adjusted prediction (GAP), which is an adaptive, nonlinear predictor. The main difference to LOCO-I is that CALIC estimates intensity gradients in vertical and horizontal directions given by

$$\begin{aligned} d_h &= |W - WW| + |N - NW| + |NE - N| \\ d_v &= |W - NW| + |N - NN| + |NE - NNE|. \end{aligned} \tag{5.2}$$

The estimated gradients are used to identify the magnitude and orientation of edges in such a way that if the vertical variation d_v is much larger than horizontal d_h, then the initial prediction is W, and in the opposite case when the horizontal variation d_h is much larger than vertical d_v, the selected initial prediction is N. In case of small or moderate differences between the estimated gradient directions, the predicted value is a weighted average of the neighboring pixels. This initial prediction is then updated by an error feedback loop in which texture contexts are used to estimate prediction error.

Our prediction templates, presented in Publications V and VI, differ from the ones in LOCO-I and CALIC. In Publication V, we have proposed four different encoding algorithms and therefore we have also used three different templates: a causal template consisting of 24 pixel elements, a causal template consisting of 12 elements, and a non-causal template consisting of 9 elements. In Publication VI, the prediction templates are developed for color images. Therefore, depending on

the color layer in question, the templates are different as the currently encoded color layer can benefit of information from previously encoded layers. In addition, due to an increasing amount of elements in template, out of which not all are significant, we have applied sparse predictor design to select the most significant elements for the prediction. The sparse design method and the level of sparsity is selected by an MDL-inspired criterion.

5.1.2 Context coding

The aim of the prediction is to exploit correlations between neighboring pixels. Unfortunately, prediction does not remove all of the correlations, which causes the resulting probability distributions to be such that they cannot be easily modeled by a single distribution. This is due to variability among neighborhoods encountered during the prediction; the prediction has different effects in smooth and constant regions compared to fast-changing regions, such as on edges. The effect of the remaining correlations to the probability distributions can be reduced by context modeling in which corresponding residuals are grouped based on the nature of their neighborhoods.

As with the prediction, the context coding approaches differ from one algorithm to another. For instance, in JPEG-LS, the context coding as follows. First, local gradients are calculated by taking the differences of the neighboring samples as follows:

$$\begin{aligned} g_1 &= NE - N \\ g_2 &= N - NW \\ g_3 &= NW - W. \end{aligned} \tag{5.3}$$

All three gradients are quantized into 9 possible values so that the number of different contexts is $9^3 = 729$. The contexts can be simplified by merging the triplets (g_1, g_2, g_3) and $(-g_1, -g_2, -g_3)$ into the same group so that the resulting total number of contexts is 365.

5.1.3 Residual coding

Residual is the difference between the value given by the model and the observation. In images, two famous approaches for encoding the residuals include adaptive arithmetic coding and Golomb-Rice coding. In Publications V and VI, adaptive arithmetic coding is applied. In Publications II and III, Golomb-Rice coding is used due to the assumption of having exponentially decaying probability distribution for residuals. Next, the both approaches, adaptive arithmetic coding and Golomb-Rice coding, are introduced.

5.1.3.1 Adaptive arithmetic coding

Arithmetic coding is a popular method for generating variable-length codes [2]. It is useful especially among small alphabets and alphabets having highly skewed

probability distributions. In addition, arithmetic coding provides an efficient approach to adaptive coding in which the symbol probabilities are changing. Among general lossless image compressors, the context-based adaptive arithmetic coding is used in CALIC. In Publications V and VI, the adaptive arithmetic coding with context modeling is applied to encode residuals.

The idea of arithmetic coding was first invented by Elias, and published in a book authored by Abramson [101]. The practical arithmetic coding algorithms were first introduced in two independent papers written by Pasco [102] and Rissanen [103]. The most well-known paper about practical arithmetic coding algorithms is by Rissanen and Langdon [104], and about the implementation by Witten et al. [105].

Arithmetic coding provides a more efficient way to encode sequences of symbols than Huffman coding which is known to be optimal for the given probability distribution of the source and results in an average codelength that is within 1 bit from the corresponding entropy. The problems of Huffman coding among small alphabets and skewed distributions stems from the fact that the difference between the average codelength and entropy can be relatively high. This difference can be improved by applying Huffman coding to the blocks of symbols. However, once the size of the block grows, the size of the extended alphabet also grows exponentially. This requires more memory, which may not be available. In addition, decoding with the resulting large alphabet is highly inefficient and time-consuming. In arithmetic coding, the sequences of symbols can be more easily encoded and there is no need to define codewords for all the possible sequences of that length. Hence, arithmetic coding is more efficient among small alphabets and skewed distributions. In addition, arithmetic coding is especially efficient on adaptive coding in which the symbol probabilities are changing.

Arithmetic coding functions as follows. Sequences of symbols are distinguished from each other by a unique tag or identifier. This tag is generated by mapping the sequence of symbols into unit interval $[0, 1)$. Since there are an infinite number of different possible tags in the unit interval, arithmetic coding is able to find a unique tag for each distinct sequence of symbols. The mapping of the sequence starts by dividing the unit interval $[0, 1)$ into non-overlapping subintervals based on the probabilities of the symbols such that each probability corresponds to the length of its associated subinterval. Depending on the first symbol in the sequence, the location of the tag is restricted to the corresponding subinterval. Then, this subinterval is divided into the same proportions as the original interval. The second symbol in the sequence determines which one of these intervals of the subinterval is chosen. In fact, the arithmetic coding procedure is locating the tag into decreasing nested subintervals based on the symbols in the sequence. As the resulting interval is disjoint from all other intervals for any other sequence, the tag can be any member within the interval.

The coding performance of the encoder is naturally defined by its ability to predict the probabilities of the symbols; the closer the predicted probabilities are to the

true ones, the more efficient the encoding is. In arithmetic coding, the prediction of the probabilities is done via determined models. The simplest models are models having zero order, in which the probability of the symbol does not take into account any contextual properties of the symbol and the probability is the plain occurrence of the symbol divided by the number of the symbols in the alphabet. In higher-order models, the probability of a symbol is based on the symbols that precede it, i.e. context. The viability of the arithmetic codes for the skewed distributions comes from the context modeling in such a way that we can define different probability distributions for different contexts. For instance, the smooth and highly variable image regions most probably have different kinds of neighborhoods and benefit from having different distributions.

In adaptive arithmetic coding, the probabilities of the symbols are updated using the increasing knowledge from the already observed and processed symbols of the sequence. As the probabilities in the encoder and decoder are updated similarly, the decoder is able to reconstruct the sequence so that the reconstructed sequence corresponds to the original one. This adaptation of the probabilities is particularly efficient on arithmetic coding due to its encoding process.

5.1.3.2 Golomb-Rice coding

The Golomb-Rice codes are known to be efficient for lossless encoding of images [106]. They have successfully been applied in LOCO-I [14], a low complexity lossless image compressor. Therefore, Golomb-Rice coding is used in Publications II and III to encode the residuals within image segmentation regions. The residuals result from the differences between the original image and the segmentation image.

The Golomb-Rice codes belong to the family of the Golomb codes which were first described by Solomon Golomb in [107]. The code is defined for non-negative integers with the assumption that the smaller the integer, the higher the probability. More specifically, Golomb codes are optimal Huffman for geometric distributions, i.e. distributions having form of $P(n) = (1-\rho)\rho^n$, where $0 < \rho < 1$. The Golomb codes have one integer parameter m that splits n, an integer to be encoded, into two parts. The first part is quotient $q = \lfloor \frac{n}{m} \rfloor$ and it is encoded using unary coding, i.e. q-length string of 1s followed by a single 0. Second part is the remainder $r = n - qm$, encoded by truncated binary encoding. In the above, $\lfloor x \rfloor$ stands for the integer part of x.

Golomb-Rice codes are the special case of the Golomb codes: $m = 2^k$ [92], which allows the remainder part r to be encoded using $\log_2 m = \log_2(2^k) = k$ bits. Thus, the total codelength for encoding an integer value n is $k+1+\lfloor n/2^k \rfloor$. The optimal value for the code parameter k is most commonly obtained by an exhaustive search [106].

In the case of a sequence also having negative integers $\{\varepsilon_i\}$, before using Golomb-Rice coding, the negative values have to be converted to non-negative integers.

One option is the following mapping

$$\gamma_i = \begin{cases} 2\varepsilon_i & \text{if } \varepsilon_i \geq 0 \\ 2|\varepsilon_i| - 1 & \text{if } \varepsilon_i < 0 \ . \end{cases} \tag{5.4}$$

For the sequence $\{\varepsilon_i\}$ having values from Laplacian distribution centered at zero, the above mapping gives a distribution close to the geometric one and the use of the Golomb-Rice code is justified [106].

5.2 Publicly available lossless image compressors

In this subsection, general lossless image compressors used as reference methods in Publications V and VI are reviewed. The approaches include Context-based, adaptive, lossless image codec (CALIC) [13], Low complexity lossless compression for image (LOCO-I) [14], the lossless version of JPEG 2000, and Lossless color image compression algorithm (LCIC) [99]. Both CALIC and LOCO-I are designed for lossless and nearly lossless coding of gray level images; albeit, some extensions to color images exist. JPEG2000 is able to compress losslessly both color and gray level images, although lossy coding with JPEG2000 is more popular. LCIC is the most recent algorithm and it is especially designed for lossless coding of color images.

CALIC and LOCO-I belong to the group of predictive lossless image compression algorithms. The general properties of such algorithms with some examples are presented above in Section 5.1. The main difference between the two methods is that LOCO-I has a much simpler predictive coder than CALIC. In addition, LOCO-I uses Golomb-Rice coding to encode residuals instead of the adaptive arithmetic coding used in CALIC. CALIC is known to compress general images more efficiently than LOCO-I [2, 14]. This can also be seen in Publications V and VI. The reason is that LOCO-I is developed to be a low complexity algorithm.

The approach of JPEG2000 to image compression is different compared to CALIC and LOCO-I. Namely, JPEG2000 is based on Discrete wavelet transformation (DWT), and therefore it belongs to the group of filtering and wavelet based compression approaches. In principle, the encoding with wavelet based compression schemes is as follows. First, the signal is decomposed using filter banks. Then, the coefficients of the filter banks are downsampled and quantized. At the end is the encoding of the coefficients. The decoding is naturally a reversal to encoding: decoding of the coded representations, upsampling, and recomposition of the signal using a synthesis filter bank.

JPEG 2000 has both lossless and lossy versions, and the properties of the lossy mode were especially of interest when invented. The JPEG 2000 standard has been created by the Joint Photographic Experts Group committee in 2000 and designated to replace the 1992-created lossy JPEG standard, which is based

on discrete cosine transform (DCT). JPEG 2000 has several advantages over JPEG of which one of the main advantages is that the bit stream of JPEG 2000 after compression is flexible and allows the bit stream to be decoded in numerous ways. For instance, the bit stream can be truncated at any point, which enables multiple resolution representations. Other advantages include superior compression performance especially at lower bit rates, artifacts being less visible, and blocking being less severe. The used coding technique is the Embedded Block Coding with Optimal Truncation (EBCOT), in which the most significant bits are encoded first followed by less significant ones. In lossless mode, all bits naturally have to be encoded and the bit stream cannot be truncated.

One of the most recent lossless image compression algorithms, LCIC [99], is similar to CALIC and JPEG-LS since it belongs to the group of predictive lossless image compressors. The main ideas behind the algorithm are hierarchical prediction and context-adaptive arithmetic coding. The difference to gray level image compressors is that LCIC is developed for color images. The LCIC compression algorithm can be summarized as follows. First, the color image is decorrelated by a reversible color transform (RCT), which is defined and used in JPEG 2000, resulting in YC_uC_v, where Y is luminance, and C_u and C_v are chrominance components. Then, the Y component is encoded by conventional lossless image compression method such as CALIC [13] or the lossless version of JPEG 2000. The chrominance images C_u and C_v are encoded using a hierarchical scheme. The proposed hierarchical scheme also allows lower pixels to be used in the prediction which is not possible with regular raster scan of the image. The hierarchical scheme is as follows. First, the image is separated into two subimages such that even rows form the first subimage X_e and odd rows form the second subimage X_o. X_e is encoded first and used in the encoding of X_o. There are two predictors for X_o: horizontal (depends on X_o) and vertical (depends on X_e), out of which the better predictor for each pixel is selected, and the direction of the prediction is transmitted as side information. Context modeling and arithmetic coding is used in the encoding of the residuals. In the experiment presented in [99], the compression performance of LCIC was better than JPEG 2000.

5.3 Lossless encoding of segmentations

In the previous section, we have reviewed publicly available lossless image compression methods for natural images. Segmentations differ from natural images, as segmentations result in constant value regions. Therefore, there are more suitable approaches to encode segmentations. One approach is based on chain codes, described in Section 4.5.1. In Publications II and III, the boundaries of the regions of the segmentation are described by parameters of ellipses. A recently proposed approach is Crack-edge-region-value (CERV) [91], which first efficiently encodes contours, i.e. boundaries of the regions, and then the mean intensity values of each region. The CERV algorithm is especially designed for compression of depth

map images which contain numerous large areas having constant value. The performance of the algorithm is shown to overtake CALIC and JPEG-LS on depth map images [91]. As the goal of segmentation is to produce large constant value areas, the CERV algorithm is also suitable for encoding segmentations. Hence, the algorithm is adopted to encode the segmentations, i.e. boundaries of the regions of segmentation and the mean intensity values of the regions, in Publications V and VI. In addition, in Publication V, the performance of the CERV algorithm is compared to CALIC and JPEG-LS and results in lower codelengths for all segmentations over one experimented histological image.

The CERV algorithm consists of two stages. However, only the first stage is needed when encoding the segmentation. Next, we will describe the first stage of the CERV algorithm, which encodes the boundaries of the regions, i.e. contours. The contours are formed by active crack-edges. A crack-edge is defined in [91] as the virtual line between two neighboring pixels, and the active crack-edge is a crack-edge in which corresponding pixels have different intensity levels. The active crack-edges can be encoded two ways: coding sequentially all crack-edges in both horizontal and vertical directions with help of bi-dimensional context coding and arithmetic coding, or using a more traditional approach, which consists of transmitting so-called anchor points and then encoding by 3OT chain-codes. The second approach is more efficient in the areas where the density of active crack-edges is low. Hence, depending on the codelengths of the approaches for an active crack-edge, the one having lower codelength is selected.

5.4 Two-phase compression of gray level histological images

The first medical image compression algorithms of this thesis are originally presented in Publication V. Altogether four different compression schemes are proposed and tested on gray-level histological images. One of the proposed approaches is encoding the image in one phase, and the other three approaches are encoding images in two-phases: a segmentation is encoded first, which is followed by encoding of the image by utilizing the segmentation. There are several motivations to encode images in two phases. The first motivation is that the resulting total codelengths could be used to rank different segmentations in the sense of minimum description length (MDL) principle, as described in the previous chapter. The other motivation is to study whether image segmentation is helping in the encoding of the image.

In our experiments, the segmentations for the two-phase algorithms are provided by the mean shift segmentation algorithm [44]. The mean shift segmentation algorithm produces segmentations on different scales by varying two parameters, spatial and range bandwidths. The mean shift algorithm is selected due to its good abilities in producing segmentations in non-homogeneous images as described

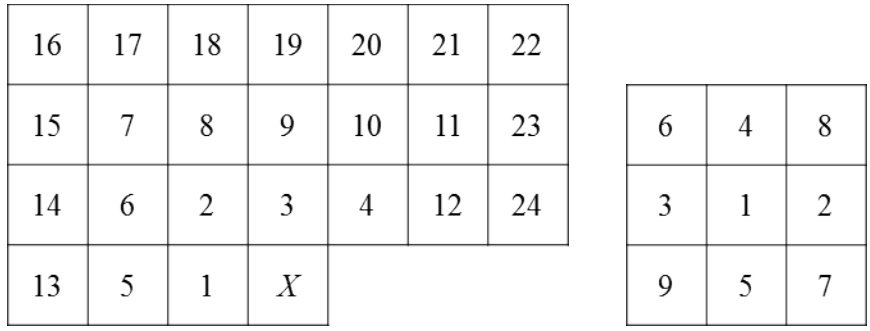
in Section 2.3.3. The other reason to use the mean shift algorithm is that the bandwidth parameters scale the segmentation so that large parameter values corresponds to under-segmentation and small values to over-segmentation. A well-known challenge of the mean shift algorithm is that the bandwidth parameters are application dependent: they need tuning to fit to local image characteristics [108]. Therefore, one motivation for selecting the mean shift algorithm to provide segmentations on multiple scale is that by a successful image segmentation ranking method, the problem of choosing the mean shift parameters could be alleviated. One combination of the mean shift segmentation and MDL has already been presented by Luo et al. [28]; and described in this thesis in Chapter 4.5.2. Luo used the mean shift algorithm to create initial over-segmentation which is followed by region merging at multiple scales by using an MDL-based criterion to decide whether or not to merge neighboring regions at each iteration. Luo created different scales by using a smoothing function within regions. The selection of the smoothing scale parameter was left to the user or to be learned from the training images. Next, the proposed lossless compression schemes for gray level images are introduced, and a summary of the experimental results is then given.

5.4.1 Two-phase lossless compression schemes for gray level images

In Publication V, we have proposed four lossless image compression algorithms which will be next reviewed. One of the proposed approaches, $\mathcal{A}_{y,P1}$, is encoding the image in one phase, and the three other approaches, $\mathcal{A}_{y,0}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$, in two phases. In all the two-phase approaches, the segmentation image is encoded first by using the CERV algorithm [91], see Section 5.3. After that, we encode the original image conditional on the segmentation. The role of the one-phase algorithm, $\mathcal{A}_{y,P1}$, is to be a reference method for two-phase algorithms $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$. Approaches $\mathcal{A}_{y,P1}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$ are linear predictive lossless image compressors, while $\mathcal{A}_{y,0}$ uses a simple additive decomposition. Residuals are encoded in all algorithms by adaptive arithmetic coding. Next, the simple additive decomposition based lossless compression algorithm, $\mathcal{A}_{y,0}$, is presented. Then, three predictive lossless algorithms $\mathcal{A}_{y,P1}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$ are introduced including the description of the templates and different predictors. Finally, we give a reasoning for using the CERV algorithm in the encoding of the segmentations.

The first introduced two-phase compression scheme, $\mathcal{A}_{y,0}$, is the only two-phase algorithm that compress segmentation regions independently once the segmentation image has been first encoded. The algorithm goes through all segmentation regions one by one and encodes on each region all the pixels that belong to the region with adaptive arithmetic coding. This approach is similar to the MDL-based segmentation algorithms presented in the previous chapter, since no spatial correlations are utilized in the encoding of the segmentation regions.

Methods $\mathcal{A}_{y,P1}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$ are linear predictive lossless image compression



(a) Causal templates T_{24} and T_{12}

(b) Non-causal template T_9

Figure 5.2: Templates used in Publication V. (a) Causal templates for predicting the current pixel X. The template T_{24} consists of 24 pixels in the causal neighborhood, and the template T_{12} consists of 12 pixels marked by 1 to 12. (b) Non-causal template is applied to already encoded segmentation image Y_{seg}. Since the current pixel is also used in prediction, the current pixel is denoted in the template as 1.

algorithms. The differences between the algorithms are the used prediction templates. An introduction to linear predictive lossless compression can be found from Section 5.1.1. In the methods, altogether three different templates are applied in the prediction. The used templates includes two causal templates having sizes of 24 and 12 pixels, and one non-causal template having size of 9 pixels. The used templates are shown in Figure 5.2. The nomination of the elements differs from the ones in Section 5.1.1 due to the large amount of elements in the template. Here, the elements are represented by numbers instead of NESW coordinates. The template having 24 elements applied to a pixel in coordinate (i, j) in an image, Y, is denoted as $T_{24}(Y(i, j))$, and the resulting regressor vector is denoted as $\phi_{1:24}(i, j)$. Correspondingly, the template having 12 elements is denoted as $T_{12}(Y(i, j))$. The non-causal template is applied to segmented image Y_{seg}, and the respective regressor vector is $\varphi_{1:9}(i, j) = T_9(Y_{seg}(i, j))$.

In the linear predictive lossless image compression algorithms $\mathcal{A}_{y,P1}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$, the prediction is done by multiplying the regressor vectors, r, with the estimated predictor vector, w, such that the predicted value for the pixel (i, j) is $\hat{Y}(i, j) = rw$. The parameters of the predictor vector, w, are estimated by the least squares estimation, and they are rounded to a precision of 10 bits. In the algorithms $\mathcal{A}_{y,P1}$ and $\mathcal{A}_{y,P2}$, the predictor vectors are the same over the whole image. In the algorithm $\mathcal{A}_{y,P3}$, the predictor vectors are context, c, specific. The algorithm specific regressor vectors are as follows. The one-phase algorithm $\mathcal{A}_{y,P1}$ uses only the template having 24 elements, T_{24}, with an addition of 1 to model the bias of the predictor. Thus, the regressor vector for predicting the pixel in location (i, j) is $[1\ \phi_{1:24}(i, j)]$. The difference between the one-phase algorithm $\mathcal{A}_{y,P1}$ and the two-phase algorithm $\mathcal{A}_{y,P2}$ is that the algorithm $\mathcal{A}_{y,P2}$ also applies

the non-causal template applied to segmented image. Therefore, the regressor vector of the algorithm $\mathcal{A}_{y,P2}$ for the pixel (i,j) is $[1\ \phi_{1:24}(i,j)\ \varphi_{1:9}(i,j)]$. The third two-phase algorithm, $\mathcal{A}_{y,P3}$, is different from the algorithms $\mathcal{A}_{y,P1}$ and $\mathcal{A}_{y,P2}$, since the predictor vectors are context c specific. The context of the pixel (i,j) is obtained by applying to image Y the causal template having 12 elements, i.e. $T_{12}(Y(i,j))$, and denoting $T_{\mathbf{z}}(Y(i,j))$ as the binary version on whether the template element belongs to the same segmentation region as the pixel (i,j) or not. Therefore, there are altogether $2^{12} = 4096$ possible contexts. The corresponding regressor vector is denoted as $\phi_{\mathbf{z}}(i,j)$. The regressor vector for the pixel (i,j) having context c is $[1\ \phi_{\mathbf{z}}(i,j)\ Y_{seg}(i,j)]$; and the predictor vector is estimated over the pixels having the same context c.

In all two-phase compression algorithms, $\mathcal{A}_{y,0}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$, the segmentation is encoded by using the CERV algorithm [91]. The algorithm was originally developed for lossless encoding of depth map images. Since both depth maps and segmentations are in most cases formed by large constant value regions, the method is also suitable for encoding segmentations. We used the CERV algorithm over a histological image with 20 segmentations resulting from the mean shift segmentation with different parameters. The CERV algorithm gave the shortest codelengths for all the segmentations when compared to the codelengths resulting from general lossless image compressors CALIC and LOCO-I.

The total codelengths for the proposed lossless image compression algorithms are next described. The total codelength of the one-phase algorithm $\mathcal{A}_{y,P1}$ consist only of the encoding of the original image Y, and is denoted as $\mathcal{L}(Y;\mathcal{A}_{y,P1})$. In the two-phase algorithms $\mathcal{A}_{y,0}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$, the segmentation is encoded first by using the CERV algorithm, and the resulting codelength is denoted as $\mathcal{L}(Y_{seg};\mathcal{A}_{s1})$. Then, the conditional images are encoded, and they are denoted as $\mathcal{L}(Y|Y_{seg};\mathcal{A}_{y,P})$, where $\mathcal{A}_{y,P}$ denotes the used algorithm. Therefore, the total codelengths of the two-phase algorithms are of the form: $\mathcal{L}(Y,Y_{seg};\mathcal{A}_{y,P}) = \mathcal{L}(Y_{seg};\mathcal{A}_{s1}) + \mathcal{L}(Y|Y_{seg};\mathcal{A}_{y,P})$.

5.4.2 Data set and experimental results

The experiments are performed on histological images from a database [109]; a detailed description of the histological images in general can be found in Chapter 3.1. The used database contains 58 H&E stained histological images of human breast cancer. The size of the color images is 896×768.

First, we tested the proposed one phase compression algorithm, $\mathcal{A}_{y,P1}$, against general lossless image compressors, CALIC and LOCO-I, over all the 58 histological images. It turned out that CALIC outperforms LOCO-I in all the images as can be seen from Figure 5.3. The proposed one-phase algorithm $\mathcal{A}_{y,P1}$ is on average resulting in the second best compression performance.

We performed a more detailed experiment over one image from the database. We

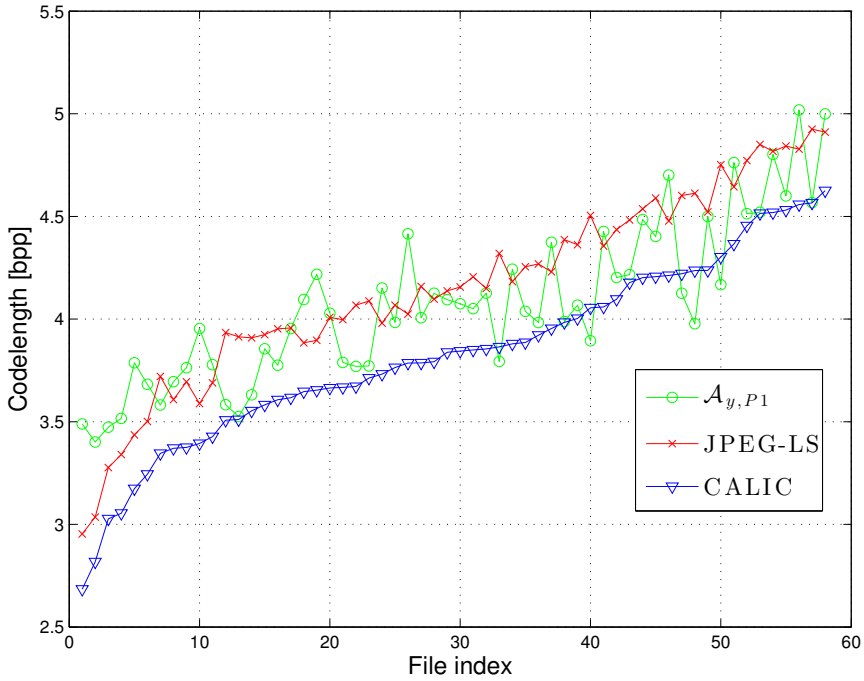


Figure 5.3: Comparison of one-phase compression algorithms: $\mathcal{A}_{y,P1}$, JPEG-LS (LOCO-I), and CALIC. The experiment is performed on 58 histological images. The figure is originally published in Publication V. © 2013 IEEE.

generated several segmentations for the image by using the mean shift algorithm. Altogether 20 segmentations were generated such that the two mean shift parameters were coupled as follows. The range bandwidth was $h_r = 2h$, and the corresponding spatial bandwidth was $h_s = 2h + 1$, when h was ranging from 1 to 20. All the proposed algorithms were compared to publicly available image compressors: CALIC, LOCO-I, and lossless JPEG 2000, which encode images in one phase.

From Figure 5.4, it can been seen that the best image compressor is CALIC. The second best is the one-phase algorithm $\mathcal{A}_{y,P1}$. JPEG2000 and LOCO-I are better than two-phase algorithms $\mathcal{A}_{y,P2}$ and $\mathcal{A}_{y,P3}$ on small mean shift parameter h values. The codelengths of the two-phase algorithms $\mathcal{A}_{y,P2}$ and $\mathcal{A}_{y,P3}$ are monotonically decreasing as the mean shift parameter h increases. The lower bound for the two algorithms is given by the one-phase algorithm $\mathcal{A}_{y,P1}$ which is reached by the algorithm $\mathcal{A}_{y,P3}$ while the codelengths of the algorithm $\mathcal{A}_{y,P2}$ stays above the codelength of the algorithm $\mathcal{A}_{y,P1}$. The two-phase algorithm $\mathcal{A}_{y,0}$ has by far the highest codelengths of all algorithms, and its performance is opposite to $\mathcal{A}_{y,P2}$ and $\mathcal{A}_{y,P3}$, since the total codelength of the algorithm $\mathcal{A}_{y,0}$ is increasing as h increases. The local minimum of the codelengths of the algorithm $\mathcal{A}_{y,0}$ is on small h values which corresponds to highly segmented images. Since with the algorithms

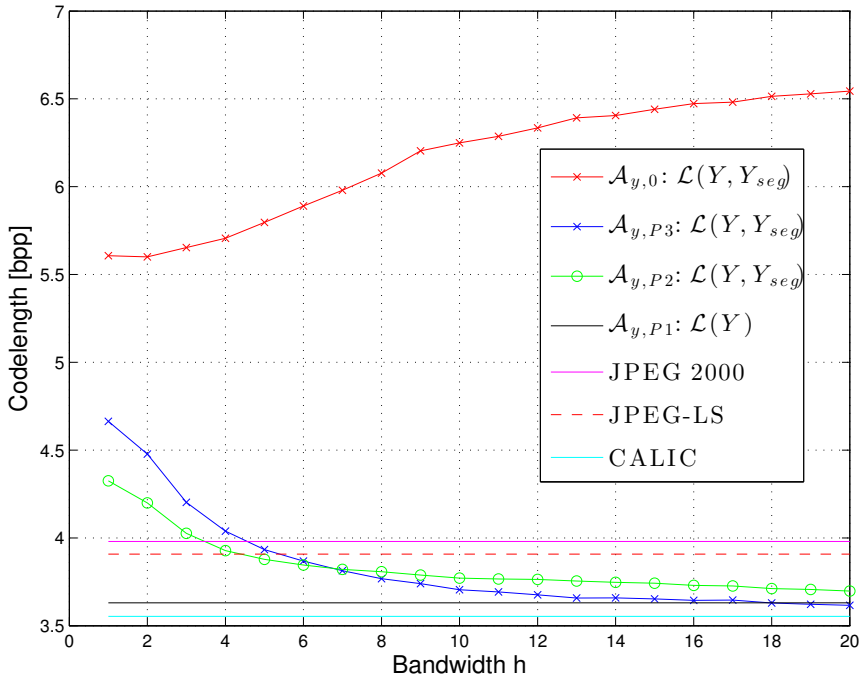


Figure 5.4: Comparison of compression for several lossless image compression algorithms over one histological image. The algorithms $\mathcal{A}_{y,0}$, $\mathcal{A}_{y,P2}$, $\mathcal{A}_{y,P3}$ are encoding images in two phases. JPEG 2000 (lossless), JPEG-LS (LOCO-I), CALIC, and $\mathcal{A}_{y,P1}$ are encoding images in one phase. The bandwidth parameter, h, is the mean shift segmentation parameter which regulates the segmentation regions: the lower the parameter, the more segmentation regions, and the higher the parameter, the fewer number of segmentation regions. The figure is originally published in Publication V. © 2013 IEEE.

$\mathcal{A}_{y,0}$, $\mathcal{A}_{y,P2}$, and $\mathcal{A}_{y,P3}$ the segmentations and encoding of segmentations are the same with respect to bandwidth parameter h, the differences between the total codelengths results from the conditional encodings of the images. The algorithm $\mathcal{A}_{y,0}$ assumes the neighboring pixels within a region to be independent, which in the case of images rarely holds. The prediction algorithms $\mathcal{A}_{y,P2}$ and $\mathcal{A}_{y,P3}$ are taking the neighboring pixel dependencies into account in the prediction step, which results in lower total codelengths. Based on the results, it seems that the proposed linear predictive compression algorithms $\mathcal{A}_{y,P2}$ and $\mathcal{A}_{y,P3}$ cannot be used in segmentation ranking.

We studied the coding performance of the algorithm $\mathcal{A}_{y,P2}$ more closely by decomposing the total codelength of the algorithm $\mathcal{A}_{y,P2}$ into the cost of encoding the segmentation and the cost of conditionally coding the image for each of the 20 segmentations, and comparing the codelengths to the one-phase algorithm $\mathcal{A}_{y,P1}$. This analysis is shown in Figure 5.5. The difference between the methods, in addition to one being one-phase and the other being a two-phase algorithm, is that the two-phase algorithm, $\mathcal{A}_{y,P2}$, also has the non-causal template used to

the already encoded segmented image. The cost of encoding the segmentation is naturally getting smaller, once the mean shift parameter value h is increasing, i.e. the number of regions is decreasing. At the same time, the cost of conditionally encoding the image by the algorithm $\mathcal{A}_{y,P2}$ is increasing. The cost of encoding the segmentation is high compared to the obtained utility in the encoding of the conditional image by using the algorithm $\mathcal{A}_{y,P2}$. Therefore, the total codelength is monotonically decreasing as h increases. It can be seen that the conditional codelengths of $\mathcal{A}_{y,P2}$ are only marginally lower than total codelength of the single-phase algorithm $\mathcal{A}_{y,P1}$, when the mean shift parameter h is lower than 5. In case of h being larger than or equal to 5, the conditional codelengths of the algorithm $\mathcal{A}_{y,P2}$ are even higher than the total codelength of the algorithm $\mathcal{A}_{y,P1}$. This shows that the non-causal template addition to prediction in the algorithm $\mathcal{A}_{y,P2}$ is increasing the codelength. One reason is that the algorithm $\mathcal{A}_{y,P1}$ has already a wide and large prediction template (24 elements), and in the algorithm $\mathcal{A}_{y,P2}$, the number of parameters in the predictor is even higher, 31 parameters. Since encoding each predictor parameter costs 10 bits, both algorithms might be improved by reducing the prediction elements to the most significant ones.

The initial 20 segmentations were generated by rough estimates for the mean shift parameters. For a finer investigation of the best segmentation according to the MDL criterion, we generated extra segmentations using a bi-dimensional grid of the mean shift parameters: the range bandwidths $h_r = 1 + 0.4i$, $i \in \{0, 1, \ldots, 10\}$ and the spatial bandwidths $h_s \in \{1, \ldots, 7\}$. The best segmentation according to the MDL criterion, i.e. having lowest total codelength by using the two-phase algorithm, $\mathcal{A}_{y,0}$, is obtained by the mean shift parameters: $h_r = 3.4$ and $h_s = 4$. These parameters are closest to $h = 1$ and $h = 2$ in the initial 20 segmentations. In a visual evaluation, the best segmentation given by the algorithm $\mathcal{A}_{y,0}$ is preserving all details of interest, including all nuclei, as shown in Figure 5.6. For instance, on mean shift segmentation with parameter $h = 5$, many of the details are lost and especially many of the nuclei are not any more detectable from the segmentation image. Thus, the algorithm $\mathcal{A}_{y,0}$ chose from the set of segmentations the one that agrees with the visual ranking.

5.4.3 Conclusions of the two-phase compression algorithms for gray-level images

Some ideas of the publication V for future development and investigations are next presented. First, the size of the template might have been too large. Each template element increases the number of predictors that needs to be encoded. A large template may contain redundant or insignificant elements that are only increasing the codelength. Second, when comparing compression results of the simple decomposition based algorithm $\mathcal{A}_{y,0}$ and the prediction based two-phase compression algorithms $\mathcal{A}_{y,P2}$ and $\mathcal{A}_{y,P3}$, the importance of removing the dependencies of neighboring pixels is clearly shown. Finally, the most important notion

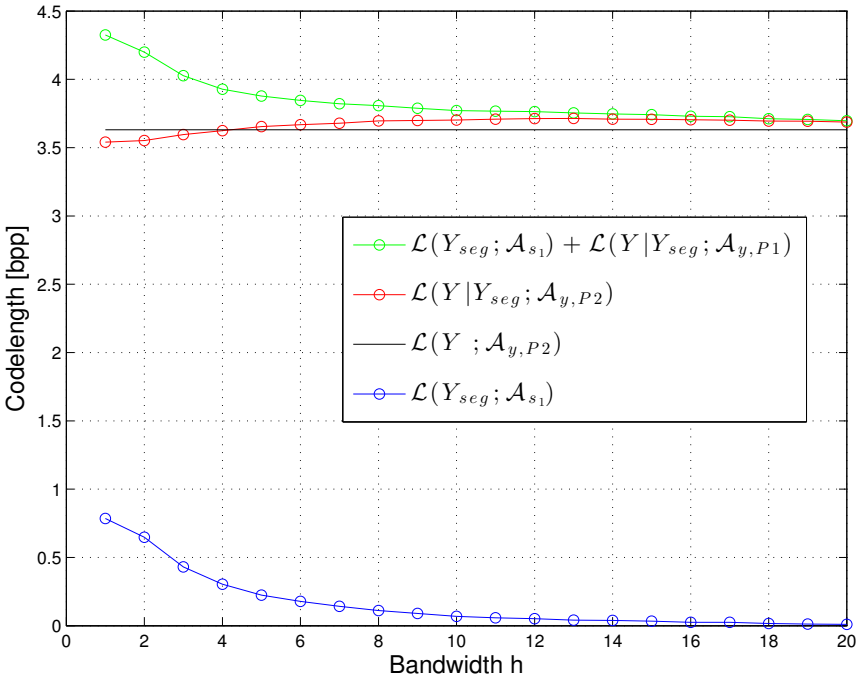
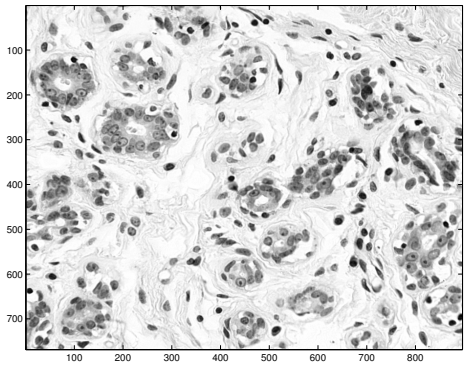


Figure 5.5: A detailed decomposition of the costs of the two-phase algorithm $\mathcal{A}_{y,P2}$, and the comparison with the one-phase algorithm $\mathcal{A}_{y,P1}$. The encoding of the segmentation images are done by the CERV algorithm, denoted by $\mathcal{A}_{y,s1}$. The bandwidth parameter, h, is the mean shift segmentation parameter which regulates the segmentation regions: the lower the parameter, the larger amount of segmentation regions there are; and the higher the parameter, the fewer number of segmentation regions. The figure is originally published in Publication V. © 2013 IEEE.

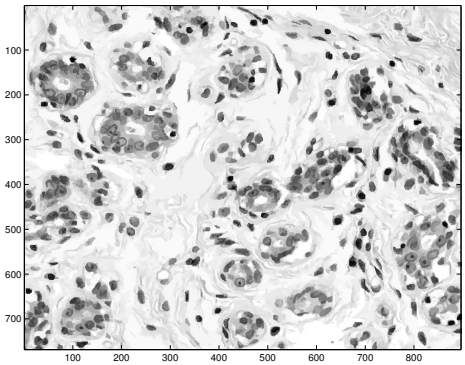
is that although predictive two-phase lossless image compressors are resulting in significantly lower codelengths, their ability to rank segmentations is arguable. Based on visual investigation, the best segmentation ranking algorithm was the one that gave the highest total codelengths, i.e. the algorithm $\mathcal{A}_{y,0}$. Therefore, to improve the compression performance, the improvement of the predictive models is important; on the other hand, it is essential to select the proper models and methods when performing statistical inference.

5.5 Lossless compression of regions-of-interest in retinal color images

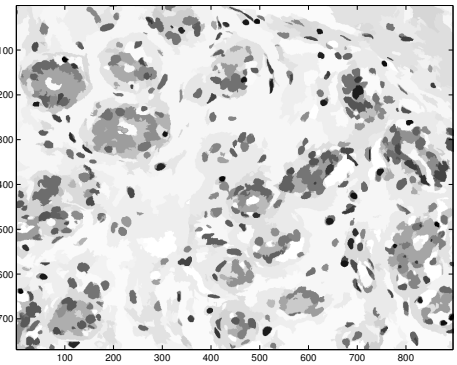
In Publication VI, we have proposed a lossless image compression algorithm for retinal color images. The proposed approach allows encoding of the vessels and the remaining part of the eye fundus to be done independently, so that depending on the application, vessels and the remaining part can be transmitted separately, once



(a) Original image



(b) The best mean shift segmentation



(c) Image segmentation with the mean shift parameter $h = 5$

Figure 5.6: Visual comparison of two mean shift segmentations. (a) Original image. (b) The best mean shift segmentation based on the total codelengths resulting the two-phase algorithm $\mathcal{A}_{y,0}$. It is obtained by the spatial and range parameters: $h_s = 4$, $h_r = 3.4$, which corresponds to h that is between 1 and 2. (c) Mean shift segmentation with parameter $h = 5$. The figures are originally published in Publication V. © 2013 IEEE.

the segmentation of the vessels is transmitted first. The approach is motivated by an emerging interest in the fields of lossless image and video compression, where encoding losslessly only the regions-of-interest has been proposed to the existing image and video compression standards [110, 111]. Among medical imaging and telemedicine applications, lossless encoding of regions-of-interest is especially important, since the sizes of medical images can be enormous, the transmission and storage of the full image might not be applicable, and the loss of quality cannot be afforded in diagnostically important regions.

The proposed algorithm is also influenced by sparse prediction and context coding of stereo color images presented in [112]. The need of sparse prediction, especially for color images, stems from the increasing number of template elements available for the prediction of the pixel values. Namely, color images consist of three layers, and in addition to the correlations between neighboring pixels, there are also

correlations between the layers which can be added to the template. Naturally, in the general gray level image compressors such as CALIC and LOCO-I, introduced in Section 5.2, these correlations between the layers are not taken into account. The correlations can be reduced by a proper color transformation, such as reversible color transform (RCT), defined in JPEG 2000. However, the transformation does not necessary remove all the correlations. Sparse predictor design allows to select the most relevant prediction template elements, and leaves out the elements that are redundant or insignificant. The sparse predictor design method, and the level of sparsity is selected by an MDL-inspired approach.

Next, the proposed sparse prediction based lossless compression scheme for color images is introduced. Then, the retinal image data set used in the experiments is described. Finally, there is a summary of the experimental results.

5.5.1 Sparse prediction based lossless compression scheme for color images

The proposed sparse prediction based region-confined lossless compression scheme is described next. First, the used prediction templates, region-confined encoding, and predictors are introduced. Then, three sparse prediction design methods are presented. We also give a review of an MDL-inspired measure, which we used to rank the sparse prediction methods. In addition, we show the comparison of the three methods over an image. After that, we describe our context coding approach for improving the coding performance. Finally, we will describe the overall codelength of the compression algorithm.

In our proposed linear predictive compression algorithm, the prediction templates are designed for color images. In our approach, an RGB image is encoded sequentially. The order of the color layers for the encoding is: the red color component, R, the green color component, G, and the blue color component, B. The prediction templates consist of the causal template of the current color component, and of the non-causal template of already encoded color components, if such exist. Therefore, the template for the red component is $[N_R W_R NW_R NE_R]$, the template for the green component is $[N_G W_G NW_G NE_G N_R W_R NW_R NE_R E_R SW_R S_R SE_R X_R]$, and the template for the blue component is $[N_B W_B NW_B NE_B N_R W_R NW_R NE_R$ $E_R SW_R S_R SE_R X_R N_G W_G NW_G NE_G E_G SW_G S_G SE_G X_G]$, where N, S, E, W correspond to north, west, east and south from the currently predicted pixel location X, as presented in Section 5.1.1, and the sub-indexes mark the respective color components.

We want to encode each segmentation region, in this case vessels and non-vessel, independently. Therefore, we apply region-confined encoding, in which the template elements, not belonging to the same region as the pixel we are predicting, are substituted by the value of the first pixel from the template that belongs to the respective region. In the case of an empty template, no template elements

belongs to the same segmentation region as the pixel we are aiming to predict. In that case, we cannot predict the pixel, and it is encoded using 24 bits.

In our approach, the linear prediction is done for each color component and each segmentation region separately. We form regressor vectors from the pixel values of the corresponding templates elements. In addition, we add 1 to each regressor vector to model the bias of the predictor. In our prediction model, the prediction for a pixel is done by multiplying the respective regressor vector with the estimated predictor vector. The predictor vectors for each color component and for each segmentation region are estimated separately by the least squares estimation. For instance, the pixel (i, j) in the red color component is predicted as $\hat{y}_R(i, j) = rw$, where $r = [y_R(i-1, j)y_R(i, j-1)y_R(i-1, j-1)y_R(i-1, j+1)1]$ is the regressor vector and w is the estimated predictor vector. The estimation for the predictor vector is obtained by the least squares: $\hat{w} = (R^T R)^{-1} R^T \bar{y}_R$, where $\bar{y}_R$ is the vector for all the values of the pixels that belong to the region under consideration and R is the regressor matrix consisting of the corresponding regressor vectors. A general view of templates and prediction among lossless image compression is presented in Section 5.1.1.

The lengths of the full predictors in red, green, and blue components are 5, 14 and 23, respectively. In our approach, each predictor coefficient is quantized to a precision of 19 bits. This also means that in our model each predictor coefficient costs 19 bits. By limiting the template elements to most significant ones, we can reduce the cost of encoding the model parameters. The most relevant and significant template elements can be chosen by sparse predictor design. A thorough tutorial to sparse predictor design methods can be found in [15]. In Publication VI, we have studied three different sparse predictor design methods, which include the greedy LS design [15], the Backward design [112], and the Hard thresholding design [15]. The first sparse design method, the greedy LS design, is adding the elements to the support of the predictors one by one such that at each stage all the remaining unused template elements are in turn tentatively added to the template. The template element that has the best performance is added to the actual template support. We have evaluated the performance of the tentative template elements by an implementable MDL-inspired criterion. The template element that results into lowest total codelength is added to the actual template support. Here, the total codelengths consists of the encoding of the prediction residuals and the encoding of the quantized predictor coefficients. The residuals are encoded by arithmetic coding with a model order 0, and the coefficients are encoded by using 19 bits for each. The second sparse design method is the Backward design [112], which utilizes the absolute values of the predictor coefficients. The idea behind the method is that the higher the absolute value is, the more important the coefficient is. Therefore, the predictor coefficients are arranged into decreasing order based on their absolute values. The ordering is done only once at the beginning of the method. Then, the template elements are added to the template support in that order, and after each addition, the

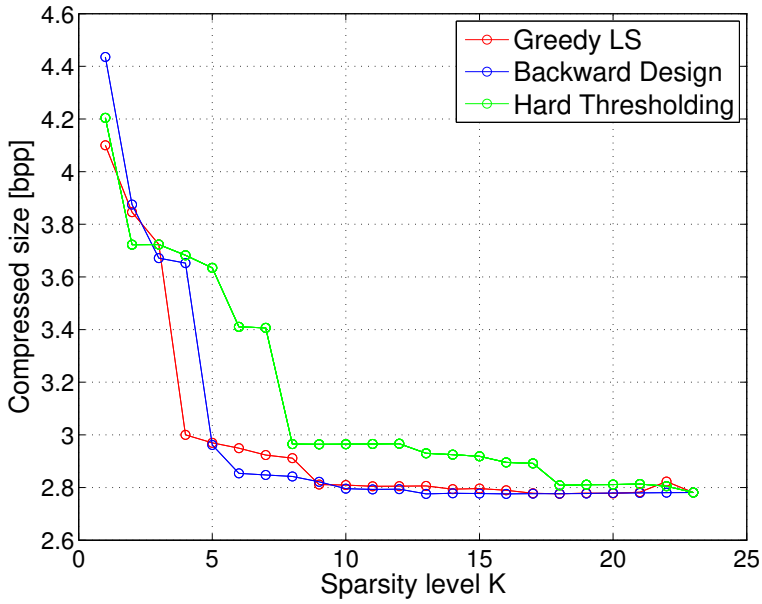


Figure 5.7: Comparison of three sparse predictor design methods for different sparsity levels K. The tested sparse predictor design methods are: Greedy LS, Backward design, and Hard thresholding. The experiment is done on 20th image of the test set in the DRIVE database. The used region is the vessel region and the color layer is the blue color component, which has the largest number of template element proposals. The figure is originally published in Publication VI. © 2014 IEEE.

MDL-based criterion is computed. The third sparse design method used is the Hard thresholding, which is the fastest algorithm. The method orders the elements of the template only once. This is done at the beginning of the method based on the absolute values of the scalar products between the regression matrix and the vector having the true values.

Each sparse design method is evaluated by the MDL-inspired criterion after each addition to the template support. The codelengths for the three methods over one retinal image with sparsity levels ranging from 1 to 23 on the blue component in the vessel region are shown in Figure 5.7. It can be seen that the second method, the Backward design, obtains lower codelengths than the other two methods most of the time. In addition, the codelengths of the first and the seconds methods are not significantly improving after the sparsity level 10. The third method, the Hard thresholding gave the worst results. Similar results were obtained for few other cases. Therefore, we decided to continue to the actual experiments with the second method, the Backward design, and we selected the sparsity levels to be 5, 10 and 10 for the red, green, and blue color components, respectively.

The coding performance of the proposed compression algorithm was improved by using context coding. We used a simple context coding approach in which

the local gradients for four directions are estimated and quantized. After several experiments, we chose to use a threshold value of 2 on each of the four gradient direction. Hence, we have a binary value from each gradient direction, and which results in 16 distinct scalar contexts. The arithmetic coding counts are initialized by transmitting the minimum and maximum values of the prediction residuals over each scalar context. For an introduction to context coding, see Section 5.1.2.

The overall compression method consists of three main parts: encoding of the segmentation, encoding of the non-vessel part, and encoding of the vessel part. The encoding of the segmentation is done by the CERV algorithm [91], introduced in Chapter 5.3. The CERV algorithm was also selected in Publication V, and it was shown therein that the algorithm gives the lowest codelengths for histological image segmentations, when compared to CALIC and LOCO-I. The codelength for encoding the segmentation is denoted as $\mathcal{L}_s$. The encodings of the non-vessel and vessel parts are done independently, and their sparse predictive compression scheme is described above. The codelengths for the non-vessel and vessel parts are denoted as $\mathcal{L}_n$ and $\mathcal{L}_v$, respectively. The resulting total codelength is the sum of the three parts, i.e. $\mathcal{L}_T = \mathcal{L}_s + \mathcal{L}_n + \mathcal{L}_v$.

5.5.2 Retinal image data set used in the experiments

Retinal images are images of eye fundus [113], designed to reveal the interior surface of the eye including retina, optic disk, macula, fovea, and vessels. Retina is a layered tissue lining located on the back inside of the eyeball. The purpose of the retina is to convert incoming light into a neural signal. The optic disk is the exit point of the ganglion cell axons from the eye. It is also the entry point for the major blood vessels. In the optic disk, there are no photoreceptor cells, rods or cones, so it is responsible for a small blind spot area. Macula is located near the center of the retinal images, and at the center of the macula is fovea, which is responsible for sharp central vision. Blood is supplied to retina by two circulations. One blood circulation is through the choroid, which is located on the outer layer of retina. The second blood circulation is the retinal circulation, which supplies the inner layer of the retina. The entry point of the retinal circulation is on the optic disk.

The main advantages of retinal images are that the images can be obtained fast and noninvasively. Retinal images provide important information about the general health of an individual via revealing the state of the blood circulation system. The diseases which can be monitored by retinal images include, for instance, cardiovascular diseases and some complications of diabetes. Thus, retinal images are highly suitable for computer assisted diagnosis (CAD) in which abnormal features can be extracted from the image and used for assessing the diagnosis. One of the main features for diagnosis in retinal images includes vessels and their segmentation. Some attributes of retinal blood vessels for diagnosis include e.g. length, width, tortuosity, and branching pattern.

Numerous vessel segmentation algorithms have been proposed for retinal images; for a survey of algorithms, see e.g. [114]. One highly competitive segmentation algorithm is based on 2D Gabor wavelets at multiple scales which is followed by a Gaussian mixture model classifier [115]. We selected that algorithm to provide segmentations for the experiments, where we evaluated the performance of the proposed lossless image compression algorithm against publicly available image compressors.

The experiments are performed on the Digital Retinal Images for Vessel Extraction (DRIVE) database [116], which is one of the databases of retinal images. The database consists of 20 images for testing. The size of the retinal images is 584×565 pixels. The eye fundus is shown in an almost circular region, with a diameter of about 540 pixels. The pixels in the retinal images are originally encoded using 8 bits for each color layer. In addition, the database provides for each image two manual vessel segmentations. The third manual segmentation for the experiments is from the database of the 2D Gabor wavelets based retinal segmentation algorithm [115].

5.5.3 Summary of the experimental results

In the experiments, we tested the proposed sparse predictive lossless image compression algorithm with eight segmentations: 5 segmentations provided by the algorithm [115], and 3 manual segmentations [115, 116], over 20 retinal test images from the DRIVE database [116]. The results were compared to publicly available lossless image compressors: CALIC [13], LOCO-I [14], LCIC [99], and lossless JPEG2000 [97, 98]. For clarity reasons, we presented for each retinal image from the proposed compression algorithm only the best and the worst performing segmentations in the sense of having the lowest and the highest total codelengths, respectively. From Figure 5.8, it can be seen that over those 20 images, in most of the images even the worst performing segmentation is giving lower codelengths than any other lossless image compressor. This means that the performance of the proposed compression algorithm is uniform and all eight segmentations provide efficient image compression results.

To compare different segmentations against their compression performance, we ordered the segmentations based on their total codelengths such that the smaller the total codelength was, the better the ranking was. It turned out that the best performing segmentation was the one provided by the segmentation algorithm with the smallest parameter value, $\sigma = 1$, and the second best with the highest parameter value, $\sigma = 20$. The manual segmentations were not the best segmentations, as one might expect. Therefore, more research on the topic is needed.

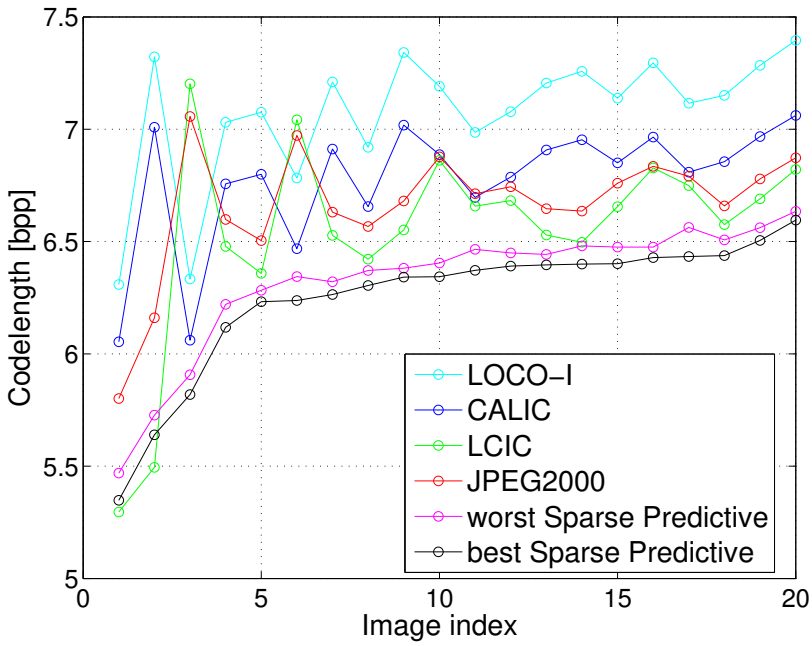


Figure 5.8: Comparison of compression for several lossless image compression algorithms over 20 retinal color images from the DRIVE database [116]. From the proposed sparse prediction based lossless image compression algorithm, we have shown for each image only the best and worst compression results out of 8 possible results. The file indexes are reordered for better visibility. The figure is originally published in Publication VI. © 2014 IEEE.

5.5.4 Discussion of the results

We have proposed an efficient sparse prediction based image compression algorithm for retinal color images. In the experiments, it performed in most of the cases better than publicly available image compressors. The algorithm allows the different image segmentation regions, in this case vessels and non-vessels, to be encoded separately, once the segmentation image is transmitted first. Therefore, the proposed algorithm provides an important approach to medical image compression and telemedicine applications, where no information on the diagnostically important image regions are allowed to be lost. In addition, the method provides an interesting approach to selecting between competing sparse predictor design methods and the level of sparsity. Namely, it was inspired by the MDL principle, and the selection was done based on the total codelengths resulting from the encoding. Before the proposed compression algorithm can be used to rank segmentations, more research is needed. As shown in Publication V, linear prediction based two-phase compression algorithms can be efficient encoders. However, their ability to rank segmentations needs to be analyzed carefully.

About the compression results of CALIC and LOCO-I, we have to admit that their

coding performance might have been improved by a proper color transformation, such as reversible color transformation (RCT). In our experiments, we applied the compression algorithms directly to the three color layers, and hence, the correlations between the color layers were not taken into account. On the other, the recently proposed lossless image compressor LCIC [99] was not better than our approach, although the approach is especially designed for color image coding.

6 Conclusions and future directions

This thesis proposed several approaches to model selection for several applications, including: signal analysis and segmentation and compression of medical images. The proposed approaches make use of parametric modeling for producing competing interpretations and representations of images and signals in specific applications. We start from heuristic approaches to model selection, which are then developed into more principled and refined ones. Namely, the developed methods belong to information theoretic based model selection approaches and are inspired by the minimum description length (MDL) principle.

In the segmentation application, the use of a parametric modeling scenario allows to define boundaries of the segmentation regions, which have pixels exhibiting similarity of their intensity or color. In the compression application, the parametric models are useful in separating the images into regions having similar types of redundancy, and applying for each such region the most suitable compression method. In both these scenarios, the MDL principle balances between the model complexity and the fitting of the data. In general, higher complex models will have a larger number of regions and more complex boundaries of the regions, involving a high model cost, but having the benefits of fitting the data better and getting smaller residuals.

One important application of this thesis has been segmentation of cell nuclei from H&E stained histological images. In histological images, both the cell nuclei and the background regions have variations, making segmentation a difficult task. In addition, the segmentation images resulting from ordinary segmentation algorithms usually contain some amount of nuclei clumps, resulting from the overlapping and touching cell nuclei that have no gradients on their borders, or the gradient is too shallow, to guide the segmentation algorithms to separate the nuclei into individual ones. We have presented a parametric setting in which the contour of a cell nucleus is assumed to be elliptical, being described by five ellipse parameters (the coordinates of the center, major and minor axes, and the angle between the major axis and the x-axis), and the representation of a clump of nuclei is given by a union of ellipses.

In Publication I, the nuclei were represented by ellipses, and the modeling goal was to find their number, shape and locations, which was accomplished by the proposed SNEF algorithm. The algorithm combined intensity and gradient information by relatively simple and fast image processing tools, such as thresholding and morphological operations. The resulting edge image was utilized for proposing numerous candidate ellipses, out of which the ellipses for the final representation were selected by a newly introduced goodness-of-fit criterion.

In Publication II, the model selection was developed differently, by resorting to a more principled approach, based on the MDL principle in the two-part coding form. Prior work on segmentation based on the MDL principle was done earlier by Kanungo et al. [27], for generic objects, where the model cost involved had particular forms, determined by their used assumptions about Gaussian distribution of the residuals and about shape encoding methods using chain codes. In our method, we have different assumptions, starting with the elliptical priors for the objects of interest and continuing with the Laplacian distributions of the residuals, which leads to the using of Golomb-Rice coding for residuals. Since the complexity of the search for the best model is very high, we chose to start by picking as possible initial solutions various ellipse configurations obtained by the SNEF algorithm, for different thresholding methods. Out of these competing interpretations, the best interpretation was selected based on the MDL-based criterion corresponding to our assumptions.

In Publication III, we studied the MDL-based selection of interpretations obtained when the original image has undergone various spatial transformations, namely smoothing and scaling, which resulted in eleven distinct interpretations. When compared to the ground truth segmentations obtained by five subjects, we found that the best interpretations based on the MDL-based criterion are among the interpretations close to the ground truths. Therefore, the proposed MDL-based criterion is applicable for selecting between competing interpretations of the cell nuclei clump, and providing solutions for the difficult image segmentation and representation problem.

In Publications V and VI, the problem of ranking different segmentations, in the sense of the best description length, has been further studied by proposing and comparing several different predictive lossless image compression algorithms. The main approach has been to consider the joint compression of the original image, and of its segmentation, such that different segmentations could be ranked based on the total codelength resulting from the compression of the joint image and segmentation. In addition, other motivations for adding the segmentation image to the task of compression have been that the segmentation may help in the encoding of an image, and that it allows lossless encoding to be applied only to the regions-of-interest.

In Publication V, four different lossless image compression algorithms were proposed and tested against general-purpose lossless image compressors. They varied

essentially in the size and shape of their prediction templates, in using of contexts coding, and in using different segmentations. The baseline algorithm, $\mathcal{A}_{y,0}$, encodes images such that no prediction templates or contexts were utilized, and the residuals of pixels of the segmentation regions were encoded using adaptive arithmetic coding. It was also the only algorithm that guaranteed region-confined encoding. Therefore, it was, in a sense, simulating models presented in Publication II and III. In Publication V, the experiments were performed on a histological image data set, and arbitrary-shaped segmentations were obtained by varying the parameters of the mean shift algorithm. In addition, instead of using ellipses for describing the contours of the regions, the segmentations were encoded by using the CERV algorithm, the state-of-the-art encoder for images having large constant regions.

In the experiments presented in Publication V, we found that the algorithms having large templates, which also covered the other regions, did not benefit from having segmentations. Their compression performance was at best as good as the corresponding algorithm that was encoding images on one-phase and without having any segmentations. The performance of the proposed two-phase compression algorithms, except the algorithm $\mathcal{A}_{y,0}$, were most of the time better than LOCO-I, and JPEG 2000. CALIC was the best compressor. The total codelengths resulting from the algorithm $\mathcal{A}_{y,0}$ were much higher than the codelengths obtained by the other compressors. In addition, the performance differentiated from the other proposed two-phase compressors since the lowest codelengths were obtained by the segmentations having a very large number of regions. The reason for these findings is that segmentations alone are not efficient enough to model all the spatial correlations present in images. Efficient approaches for modeling spatial correlations are provided by predictive templates (local) and contexts (global), and, at the same time, they perform well even without using segmentations for representing regions. Therefore, when designing models, one should bear in mind not to over-model so that the model selection would select between representative models. In this application, we were interested in ranking segmentations based on their ability to describe image regions having similar intensities, and therefore, the results of the algorithm $\mathcal{A}_{y,0}$ can be justified.

In color image compression, the prediction templates may contain a large number of elements, since already processed color layers can also be used in the prediction. Therefore, it is essential to restrict the used template elements to the most relevant ones; those can be selected by sparse prediction design methods. In Publication VI, we have applied an MDL-based criterion to select between three different sparse prediction design methods. In addition, we have used region-confined encoding that allows segmentation regions to be compressed and transmitted independently one from another, once the segmentation image has been transmitted first. In the experiments presented in Publication VI, we have applied our sparse prediction and region-confined color image compression algorithm to retinal images. In retinal images, the segmentations are separating images into vessel and non-vessel

regions. The segmentations were obtained by one of the best algorithms for vessel segmentation. In addition, we utilized some manual segmentations. It turned out that our approach was most of the time the best performing image compressor when compared to publicly available general-purpose image compressors, such as CALIC, LOCO-I, JPEG 2000, and LCIC.

In 1D time series data, linear predictive models are usually the autoregressive (AR) models, for which the model order is defined by the number of previous time samples used in prediction. In Publication IV, a recent implementation of MDL, namely the sequentially normalized maximum likelihood (SNML) model, was applied to signal analysis and interpretation. Since SNML is especially appropriate for time series analysis, we have proposed a signal change detection algorithm which combines SNML with autoregressive (AR) models. In the algorithm, the time series signal is split into smaller segments; and for each segment we minimized the SNML and AR based model selection criterion by an exhaustive search. In the experiments, it turned out that the AR model orders selected by the criterion are not good estimates for interpreting the complexity of the signal, while the corresponding values of the minimized model selection criterion seem to correlate with signal complexity, and can be used to detect changes in signals. This detected correlation between the MDL-based criterion and signal complexity coincides with the idea of the MDL principle: the codelength for encoding a signal stays rather constant if there are no changes within the phenomena producing the signal, while a change in the signal (complexity) causes the codelength for encoding the signal to also change.

The results of this thesis could be of interest to researchers working among medical image analysis, processing and transmission, and from the broader perspective, for those working on other similar image analysis and signal analysis tasks. In addition, this thesis providing numerous model selection techniques, most of them using principled MDL-based solutions, could be relevant for people working in analyzing and modeling empirical data.

Some of the ideas presented in this thesis, have already been further developed. Namely, the idea of selecting sparse predictors for image compression based on MDL presented in Publication VI has been further developed and studied in more detail on plenoptic images by Helin et al. [117].

Other possible future research directions are described next. A general problem with the detection of individual objects from clumps is that it is time-consuming. Using a parametric approach, as presented in this thesis, on which the contour of regions are described by ellipses such that one ellipse represents one nucleus, the number of parameters for representing a clump increases rapidly as the number of objects within the clump increases. The proposed SNEF algorithm can detect several ellipses, we have proposed a goodness-of-fit based criterion for selecting ellipses for the final representation, and the MDL-based criterion is able to select between competing interpretations of the clump. However, there is still a need

for an algorithm that would efficiently optimize the parameters of an ellipse that belongs to a clump independently of the other ellipses, so that parallel processing could be used for the task. In this thesis, a cell nucleus was selected to be represented by ellipses, due to ellipses having a moderate level of complexity and to nuclei resembling ellipses, as already discussed in the thesis. However, other possible parameterizations could be tested in the future, e.g. splines or other 2D geometrical structures.

In Publication V, the region-confined encoding is guaranteed only on one of the four proposed lossless image compression algorithms. In addition, in some of the proposed algorithms, the prediction templates are wide and may contain unnecessary elements. As shown in the experiments presented in Publication VI, the region-confined encoding with sparse prediction may improve the compression performance of the compressors presented in Publication V.

In Publication IV, we applied the SNML based model selection criterion for signal change detection. In the current version of SNML, the values of the criterion for the signal segments need to be calculated recursively so that for each signal segment, the calculation has to be started from the beginning. That is time-consuming and prevents its use in real-time applications. Therefore, one target for future development could be studying approaches to update the criterion once a sample or samples are removed from the beginning of the signal segment.

As a conclusion, this thesis has presented and studied several new approaches for image segmentation, compression and interpretation. We have shown the importance of model selection for selecting between competing representative models. In this thesis, the main contributions are inspired by the MDL principle, which has allowed ranking different parametric image representations, selecting between sparse predictor design methods in image compression, and providing interpretations for time series signals to be used in signal change detection.

Bibliography

[1] R. Gonzales and R. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, New Jersey: Pearson Education, Inc., 2008.

[2] K. Sayood, *Introduction to Data Compression*, 2nd ed. Morgan Kaufmann, 2000.

[3] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Transactions on Image Processing*, vol. 2, pp. 176–201, 1993.

[4] S. Kumar, S. Ong, S. Ranganath, T. Ong, and F. Chew, "A rule-based approach for robust clump splitting," *Pattern Recognition*, vol. 39, pp. 1088–1098, 2006.

[5] P. Hough, "Method and means for recognizing complex patterns," Patent U.S. 3 069 654, December, 1962.

[6] R. Duda and P. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, pp. 11–15, 1972.

[7] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[8] H. Talbot and B. Appleton, "Elliptical distance transforms and object splitting," in *Proceedings of the VIth International Symposium on Mathematical Morphology*, Sydney, Australia, April 2002, pp. 229–240.

[9] A. Garrido and N. Perez de la Blanca, "Applying deformable templates for cell image segmentation," *Pattern Recognition*, vol. 3, no. 5, pp. 821–832, 2000.

[10] K. Hahn, Y. Han, and H. Hahn, "Extraction of partially occluded elliptical objects by modified randomized hough transform," in *KI 2007: Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, J. Hertzberg, M. Beetz, and R. Englert, Eds. Springer Berlin Heidelberg, 2007, vol. 4667, pp. 323–336. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-74565-5_25

[11] D. K. Prasad, M. K. Leung, and S.-Y. Cho, "Edge curvature and convexity based ellipse detection method," *Pattern Recognition*, vol. 45, pp. 3204–3221, 2012.

[12] X. Bai, C. Sun, and F. Zhou, "Splitting touching cells based on concave points and ellipse fitting," *Pattern Recognition*, vol. 42, pp. 2434–2446, 2009.

[13] X. Wu and N. Memo, "Context-based, adaptive, lossless image coding," *IEEE Transactions on Communications*, vol. 45, pp. 437–444, 1997.

[14] M. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Transactions on Image Processing*, vol. 9, pp. 1309–1324, August 2000.

[15] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing.* Springer, 2010.

[16] P. A. Lachenbruch and M. R. Mickey, "Estimation of error rates in discriminant analysis," *Technometrics*, vol. 10, pp. 1–11, 1968.

[17] B. Efron, "Bootstrap methods: Another look at the jackknife," *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979.

[18] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Proceeding of the Second International Symposium on Information Theory*, Budapest, 1973, pp. 267–281.

[19] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[20] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, September 1978.

[21] J. Rissanen, "Fisher information and stochastic complexity," *IEEE Transactions on Information Theory*, vol. 42, pp. 40–47, 1996.

[22] A. Kolmogorov, "Three approaches to the quantitative definition of information," *Problems of Information Transmission*, vol. 1, pp. 3–11, 1965.

[23] Y. Shtarkov, "Universal sequential coding of single messages," *Problems of Information Transmission*, vol. 23, no. 3, pp. 3–17.

[24] T. Roos and J. Rissanen, "On sequentially normalized maximum likelihood models," in *Proceedings of the Workshop on Information Theoretic Methods in Science and Engineering*, 2008.

[25] J. Rissanen, T. Roos, and P. Myllymäki, "Model selection by sequentially normalized least squares," *Journal of Multivariate Analysis*, vol. 101, pp. 839 – 849, 2010.

[26] Y. Leclerc, "Constructing simple stable descriptions for image partitioning," *International Journal of Computer Vision*, vol. 3, pp. 73 – 102, 1989.

[27] T. Kanungo, B. Dom, W. Niblack, and D. Steele, "A fast algorithm for MDL-based multi-band image segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*. Seattle: IEEE, June 1994, pp. 609 – 616.

[28] Q. Luo and T. Khoshgoftaar, "Unsupervised multiscale color image segmentation based on MDL principle," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2755 – 2761, September 2006.

[29] P. Sahoo, S. Soltani, and A. Wong, "A survey of thresholding techniques," *Computer Vision, Graphics, and Image Processing*, vol. 41, no. 2, pp. 233 – 260, February 1988.

[30] M. Sezgin and B. Sankur, "Survey over thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, pp. 146 – 168, 2004.

[31] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, pp. 62 – 66, 1979.

[32] M. Hu, X. Ping, and Y. Ding, "Automatic cell nucleus detection using improved snake," in *International Conference on Image Processing*, Singapore, October 2004, pp. 2737 – 2740.

[33] S. Beucher and C. Lantuéjoul, "Use of watersheds in contour detection," in *Proceedings of the International Workshop on Image Processing, Real-Time Edge and Motion Detection*, Rennes, France, September 1979.

[34] T. Hong and A. Rosenfeld, "Compact region extraction using weighted pixel linking in a pyramid," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 222 – 229, 1984.

[35] A. Leonardis, A. Gupta, and R. Bajcsy, "Segmentation of range images as the search for geometric parametric models," *International Journal of Computer Vision*, vol. 14, pp. 253 – 277, 1995.

[36] S. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and bayes/MDL for multiband image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 884 – 900, 1996.

[37] T.-Y. Philips, A. Rosenfeld, and A. Sher, "O(log n) bimodality analysis," *Pattern Recognition*, vol. 22, pp. 741 – 746, 1989.

[38] V. Grau, A. Mewes, M. Alcaniz, R. Kikinis, and S. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Transactions on Medical Imaging*, vol. 23, pp. 447–458, 2004.

[39] H. Nguyen and Q. Ji, "Improved watershed segmentation using water diffusion and local shape priors," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2006, pp. 985–992.

[40] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, pp. 32–40, 1975.

[41] E. Parzen, "On estimation of a probability density function and mode," *Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[42] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: John Wiley & Sons, 2001.

[43] C. M. Bishop, *Pattern Recognition and Machine Learning.* New York: Springer, 2006.

[44] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[45] D. Comaniciu and P. Meer, "An algorithm for data-driven bandwidth selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 281–288, 2003.

[46] M. Gurcan, L. Boucheron, A. Can, A. Madabhushi, N. Rajpoot, and B. Yener, "Histopathological image analysis: A review," *IEEE Reviews in Biomedical Engineering*, vol. 2, pp. 147–171, 2009.

[47] W. Allsbrook and e. a. K.A. Mangold, "Interobserver reproducibility of gleason grading of prostate carcinoma: General pathologist," *Hum. Path.*, vol. 32, no. 1, pp. 81–88, 2001.

[48] V. Korde, H. Bartels, J. Ranger-Moore, and J. Barton, in *Proceedings of the European Conference on Biomedical Optics.*

[49] S. Petushi, F. Garcia, M. Haber, C. Katsinis, and A. Tozeren, "Large-scale computations on histological images reveal grade-differentiating parameters for breast cancer," *BMC Medical Imaging*, vol. 6, no. 14, 2006.

[50] S. Naik, S. Doyle, S. Agner, A. Madabhushi, M. Feldman, and J. Tomaszewski, "Automated gland and nuclei segmentation for grading of

prostate and breast cancer histopathology," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano To Macro, ISBI*, May 2008, pp. 284 – 287.

[51] H. Kong, M. Gurcan, and K. Belkacem-Boussaid, "Partitioning histopathological images: An integrated framework for supervised color-texture segmentation and cell splitting," *IEEE Transactions on Medical Imaging*, vol. 30, no. 9, pp. 1661 – 1677, September 2011.

[52] G. N. Papanicolaou, "A new procedure for staining vaginal smears," *Science*, vol. 95, no. 2469, pp. 438 – 439, 1942.

[53] M. E. Plissiti and C. Nikou, "Overlapping cell nuclei segmentation using a spatially adaptive active physical model," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 4568 – 4580, 2012.

[54] C. Wählby, I.-M. Sintorn, F. Erlandsson, G. Borgefors, and E. Bengtsson, "Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections," *Journal of Microscopy*, vol. 215, pp. 67 – 76, 2004.

[55] S. Beucher and L. Vincent, "Introduction aux outils morphologiques de segmentation," in *Traitement d'image en microscopie a balayage et en microanalyse par sonde electronique*, Paris, France, March 1990.

[56] M. Faessel and F. Courtois, "Touching grain kernels separation by gap-filling," *Image Analysis & Stereology*, vol. 28, pp. 195 – 203, 2009.

[57] H. Wu, J. Barba, and J. Gil, "A parametric fitting algorithm for segmentation of cell nuclei," *IEEE Transactions on Biomedical Engineering*, vol. 45, pp. 400 – 407, 1998.

[58] M. Fornaciari, A. Prati, and R. Cucchiara, "A fast and effective ellipse detector for embedded vision applications," *Pattern Recognition*, vol. 47, pp. 3693 – 3708, 2014.

[59] A. Y.-S. Chia, S. Rahardja, D. Rajan, and M. Leung, "A split and merge based ellipse detector with self-correcting capability," *IEEE Transaction on Image Processing*, vol. 20, pp. 1991 – 2006, July 2011.

[60] P. Thévenaz, R. Delgado-Gonzalo, and M. Unser, "The ovuscule," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 382 – 393, February 2011.

[61] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square of ellipse," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, pp. 476 – 480, 1999.

[62] S. J. Ahn, W. Rauh, and H.-J. Warnecke, "Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola," *Pattern Recognition*, vol. 34, pp. 2283 – 2303, 2001.

[63] J. Porrill, "Fitting ellipses and predicting confidence envelopes using a bias corrected Kalman filter," *Image and Vision Computing*, vol. 8, no. 1, pp. 37 – 41, 1990.

[64] T. Ellis, A. Abbood, and B. Brillault, "Ellipse detection and matching with uncertainty," *Image and Vision Computing*, vol. 10, no. 2, pp. 271 – 276, 1992.

[65] P. Rosin, "A note on the least squares fitting of ellipses," *Pattern Recognition Letters*, vol. 14, pp. 799 – 808, 1993.

[66] R. Haralick and L. Shapiro, *Computer and Robot Vision*. Addison-Wesley, 1992.

[67] Y. Xie and Q. Ji, "A new efficient ellipse detection method," in *the Proceedings of 16th International Conference on Pattern Recognition*, 2002, pp. 957 – 960.

[68] Z.-Y. Liu and H. Qiao, "Multiple ellipses detection in noisy environments: A hierarchical approach," *Pattern Recognition*, vol. 42, pp. 2421 – 2433, 2009.

[69] T. Cover and J. Thomas, *Elements of Information Theory*. John Wiley & Sons, 1991.

[70] D. Huffman, "A method for the construction of minimum-redundancy codes," in *the Proceedings of the I.R.E.*, September 1952, pp. 1098 – 1101.

[71] J. Rissanen, "MDL denoising," *IEEE Transactions on Information Theory*, vol. 46, pp. 2537 – 2543, 2000.

[72] T. Roos, P. Myllymäki, and J. Rissanen, "MDL denoising revisited," *IEEE Transactions on Signal Processing*, vol. 57, pp. 3347 – 3360, 2009.

[73] P. Kontkanen, P. Myllymäki, W. Buntine, J. Rissanen, and H. Tirri, "An MDL framework for data clustering," in *Advances in Minimum Description Length: Theory and Applications*, P. Grünwald, I. Myung, and M. Pitt, Eds. The MIT Press, 2005.

[74] G. Korodi, I. Tabus, J. Rissanen, and J. Astola, "Dna sequence compression based on the normalized maximum likelihood model," *IEEE Signal Processing Magazine*, vol. 24, pp. 47 – 53, 2007.

[75] P. Grünwald, *The Minimum Description Length Principle*. MIT Press, 2007.

[76] P. Grünwald, I. Myung, and M. Pitt, *Advances in Minimum Description Length: Theory and Applications*. MIT Press, 2005.

[77] J. Rissanen, *Optimal Estimation of Parameters*. Cambridge University Press, 2012.

[78] J. Rissanen, *Information and Complexity in Statistical Modeling*. Springer, 2007.

[79] M. Li and P. Vitanyi, *An Introduction to Kolmogorov Complexity and Its Applications*, 2nd ed. Springer-Verlag, 1997.

[80] G. Chaitin, "On the length of programs for computing finite binary sequences: Statistical considerations," *Journal of the ACM*, vol. 16, pp. 145 – 159, 1969.

[81] R. Solomonoff, "A formal theory of inductive inference. Part I," *Information and Control*, vol. 7, pp. 1 – 22, 1964.

[82] R. Solomonoff, "A formal theory of inductive inference. Part II," *Information and Control*, vol. 7, pp. 224 – 254, 1964.

[83] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Transactions on Information Theory*, vol. 44, pp. 2743 – 2760, 1998.

[84] J. Rissanen, "Stochastic complexity," *Journal of the Royal Statistical Society, Series B*, vol. 49, pp. 223 – 239, Discussion: pp. 252 – 265, 1987.

[85] J. Rissanen and T. Roos, "Conditional NML universal models," in *Proceedings of the Information Theory and Applications Workshop (ITA-07)*, 2007, pp. 337 – 341.

[86] C. Wei, "On predictive least squares principles," *The Annals of Statistics*, vol. 20, no. 1, pp. 1 – 42, 1992.

[87] J. Rissanen, "A predictive least-squares principle," *IMA Journal of Mathematical Control and Information*, vol. 3, pp. 211 – 222, 1986.

[88] R. Plackett, "Some theorems in least squares," *Biometrika*, vol. 37, pp. 149 – 157, 1950.

[89] T. Kohonen, *Self-Organizing Maps*, the 3rd. extended ed. Springer, 2001.

[90] J. Rissanen, "A universal prior for integers and estimation by minimum description length," *The Annals of Statistics*, vol. 11, pp. 416 – 431, 1983.

[91] I. Tabus, I. Schiopu, and J. Astola, "Context coding of depth map images under the piecewise-constant image model representation," *IEEE Transactions on Image Processing*, vol. 22, pp. 4195 – 4210, November 2013.

[92] R. Rice, "Some practical universal noiseless coding techniques," JPL Publication 79-22, Pasadena, CA: Jet Propulsion Laboratory, Tech. Rep., March 1979.

[93] J. Ward and D. Cok, "Resampling algorithms for image resizing and rotation," in *the Proceedings of the SPIE Digital Image Processing Applications*, 1989, pp. 260 – 269.

[94] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 6, pp. 1153 – 1160, 1981.

[95] L. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, pp. 297 – 302, 1945.

[96] J. Mueller, M. Werner, and M. Stolte, "Barrett's esophagus: Histopathologic definitions and diagnostic criteria," *World Journal of Surgery*, vol. 28, pp. 148 – 154, 2004.

[97] D. Taubman and M. Marcellin, "JPEG2000: standard for interactive imaging," *Proceedings of the IEEE*, vol. 90, pp. 1336 – 1357, 2002.

[98] D. Taubman and M. Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice*, ser. The Springer International Series in Engineering and Computer Science. Springer US, 2002.

[99] S. Kim and N. Cho, "Hierarchical prediction and context adaptive coding for lossless color image compression," *IEEE Transactions on Image Processing*, vol. 23, no. 1, pp. 445 – 449, Jan 2014.

[100] S. A. Martucci, "Reversible compression of HDTV images using median adaptive prediction and arithmetic coding," in *Proceedings of the IEEE International Symposiom on Circuits and Systems*, 1990, pp. 1310 – 1313.

[101] N. Abramson, *Information Theory and Coding*. New York: McGraw-Hill, 1963.

[102] R. Pasco, "Source coding algorithms for fast data compression," Ph.D. dissertation, Standord University, 1976.

[103] J. Rissanen, "Generalized Kraft inequality and arithmetic coding," *IBM Journal of Research and Development*, vol. 20, pp. 198 – 203, 1976.

[104] J. Rissanen and G. Langdon, "Arithmetic coding," *IBM Journal of Research and Development*, vol. 23, pp. 149 – 162, 1979.

[105] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, pp. 520 – 540, 1987.

[106] M. Weinberger, G. Seroussi, and G. Sapiro, "LOCO-I: A low complexity, context-based, lossless image compression algorithm," in *Proceedings of the IEEE Data Compression Conference*. Snowbird, Utah: IEEE, March-April 1996.

[107] S. Golomb, "Run-length encodings," *IEEE Transactions on Information Theory*, vol. 12, pp. 399 – 401, 1966.

[108] D. Comaniciu, V. Ramesh, and P. Meer, "The variable bandwidth mean shift and data-driven scale selection," 2001.

[109] E. D. Gelasca, J. Byun, B. Obara, and B. Manjunath, "Evaluation and benchmark for biological image segmentation," in *IEEE International Conference on Image Processing*. IEEE, Oct 2008, pp. 1816 – 1819.

[110] H. Chen, G. Braeckman, S. Satti, P. Schelkens, and A. Munteanu, "HEVC-based video coding with lossless region of interest for tele-medicine applications," in *Proceedings of 20th International Conference on Systems, Signals and Image Processing (IWSSIP)*, Bucharest, July 2013, pp. 129 – 132.

[111] V. Sanches, F. Auli-Llinas, J. Bartrina-Rapesta, and J. Serra-Sagrista, "HEVC-based lossless compression of whole slide pathology images," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Atlanta, GA, December 2014, pp. 297 – 301.

[112] I. Tabus and P. Astola, "Sparse prediction for compression of stereo color images conditional on constant disparity patches," in *Proceedings of 3DTV-Con 2014*, Budapest, July 2014, pp. 1 – 4.

[113] M. Abramoff, M. Garvin, and M. Sonka, "Retinal imaging and image analysis," *IEEE Reviews in Biomedical Engineering*, vol. 3, pp. 169 – 208, December 2010.

[114] M. Fraz, P. Remagnino, A. Hoppe, B. Uyyanonvara, A. Rudnicka, C. Owen, and S. Barman, "Blood vessel segmentation methodologies in retinal images – A survey," *Computer Methods and Programs in Biomedicine*, vol. 108, pp. 407 – 433, 2012.

[115] J. Soares, J. Leandro, R. Cesar, H. Jelinek, and M. Cree, "Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification," *IEEE Transactions on Medical Imaging*, vol. 25, no. 9, pp. 1214 – 1222, 2006.

[116] J. Staal, M. Abramoff, M. Niemeijer, M. Viergever, and B. van Ginneken, "Ridge based vessel segmentation in color images of the retina," *IEEE Transactions on Medical Imaging*, vol. 23, pp. 501 – 509, 2004.

[117] P. Helin, P. Astola, B. Rao, and I. Tabus, "Minimum description length sparse modeling and region merging for lossless plenoptic image compression," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 1146 – 1161, October 2017.

Publications

Publication I

J. Hukkanen, A. Hategan, E. Sabo, and I. Tabus, "Segmentation of cell nuclei from histological images by ellipse fitting," in *Proceedings of the 18th European Signal Processing Conference (EUSIPCO-2010)*, Aalborg, Denmark, August 2010, pp. 1219 – 1223.

Publication II

J. Hukkanen, E. Sabo, and I. Tabus, "Representing clumps of cell nuclei as unions of elliptic shapes by using the MDL principle," in *Proceedings of the 19th European Signal Processing Conference (EUSIPCO-2011)*, Barcelona, Spain, August 2011, pp. 1010 – 1014.

Publication III

J. Hukkanen, E. Sabo, and I. Tabus, "MDL based structure selection of union of ellipse models for scaled and smoothed histological images," *Advances in Intelligent Control Systems and Computer Science*, Springer-Verlag Berlin Heidelberg, pp. 77 – 89, 2013.

Publication IV

J. Hulkkonen and J. Heikkonen, "A minimum description length principle based method for signal change detection in machine condition monitoring," in *Proceedings of the 19th International Conference on Pattern Recognition*, Tampa, Florida, December 2008, pp. 1–4.

Publication V

I. Tabus, J. Hukkanen, and I. Schiopu, "Two-phase compression of histological images with MDL ranking of segmentation images," in *Proceedings of the 19th International Conference on Control Systems and Computer Science*, Bucharest, Romania, May 2013, pp. 331 – 338.

Publication VI

J. Hukkanen, P. Astola, and I. Tabus, "Lossless compression of regions-of-interest from retinal images," in *Proceedings of the 5th European Workshop on Visual Information Processing (EUVIP2014)*, Paris, France, December 2014, pp. 1 – 6.

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-4142-1
ISSN 1459-2045